Attachments

Attach all images to this page and link/display them using this notation:

Inline display: !Attachments^filename.png!
Link: [OUG:Attachments^filename.png]

Onyx Documentation Home

Onyx 1.10 Documentation

Onyx User Guide

The Onyx User Guide is for anyone who uses the Onyx user interface--for any purpose. This includes the three types of Onyx user:

- Data Collectors (who might be nurses, laboratory technologists, interviewers, and so on)
- Participant Managers (who have permission to do some administrative tasks in addition to those of a data collector)
- System Administrators (this guide only covers the tasks they can do in the Onyx user interface--create users, purge and export data; other tasks they do like installing and configuring Onyx are covered in other documents)

Onyx Customization & Configuration Guide

The Onyx Customization & Configuration Guide is mainly for Onyx "customizers"--programmers who need to prepare a customized version of Onyx for a particular organization. The result of doing the procedures in this guide will be a war file that can be installed on a server (as explained in the Onyx Installation Guide).

This guide is also for system administrators who can use certain procedures in this guide fine-tune the Onyx configuration after the initial installation.

Onyx Installation & Administration Guide

The Onyx Installation & Administration Guide is for "installers"---the people who will install Onyx for the first time at a site. Usually this will be a system administrator who sets up Onyx from scratch.

Onyx Customization & Configuration Guide

Contents of this Guide

- Introduction
- Setting Up the Environment
- Customizing the Global Configuration
- Configuring Interview Stages
- Customizing the Consent Stage
- Building Custom Questionnaires
- Configuring Physical Measurement Stages
- Defining a Report on Physical Measurements
- Configuring Sample Collection Stages
- Configuring Logs for Experimental Conditions and Instrument Calibration
- Configuring Data Export and Purge
- Generating a Custom War File

Introduction

Onyx can be customized to meet the needs of any organization that is collecting data from participants for a medical research study. Customizing Onyx mainly involves creating custom questionnaires and configuring Onyx to collect the physical measurements and biosamples required by the study.

Planning Your Customization

To the Organization that is Customizing Onyx

We recommend that you plan your customization of Onyx using this spreadsheet. The completed spreadsheet will server as a guideline for the programmer who will carry out the customization of Onyx for your organization.

Item	Explanation	What you need to do
Enrollment mode	Onyx can manage participants with appointments or walk-in participants or both.	Decide which enrollment mode(s) you will use.
Participant information fields	Before the consent stage, Onyx can capture some identification information about the participant. This information is either provided with the appointment list or obtained when the participant is received. Mandatory fields are: participant ID, name, and DOB. Other fields are configurable. Onyx does not assign participant IDs. The study does. We need to know the format of your participant IDs so Onyx can validate them when participants are received. Participant IDs can be on barcode labels.	Define a list of participant information fields. For each field, define: • name (to be displayed onscreen) • variable name • type (date, list, and so on) • validation rule
Stage ordering	Onyx stages (consent, questionnaires, physical measurements, biosample collection) can be listed onscreen in any order that is convenient for the study.	Define a list of the stages in an interview—in the order that you want them to appear onscreen. For each stage, define: • name of the stage as it should appear onscreen • any stages that must be completed before the current stage can be started (and any data that the current stage must receive from those stage(s)) • any stages that could be contraindicated after the current stage is completed (and any data that the current stage must send to the dependent stage) • whether or not the stage can be skipped
Consent form	The participant's signature can be obtained in various ways: Manual, electronic, or both	Decide which signature modes will you use. If you will accept consent electronically, prepare the consent form in PDF format.
Questionnaire stages	One or more questionnaires can be defined. The questions in a questionnaire should be grouped into pages (questions that you want to appear on the same screen). Questions can also be grouped into sections (by topic). You can define "skip patterns"—whether or not to display certain questions based on answer(s) to previous question(s). A skip pattern can be for a page (hide some questions in the page initially and only display them if the participant gives a particular answer to another question in the page). A skip pattern can also be for a section (do not display section about smoking if participant is a non-smoker). A questionnaire can be displayed in different modes: • standard screen • touchscreen (suitable for self-administered questionnaires) • both	For each questionnaire, define: • a list of questions (grouped by section, if sections used) • display mode(s) • skip patterns For each question, define: • text to display onscreen • answer type (e.g single choice, multiple choice, open text) • variable name(s) • category names (answers to display onscreen) • category codes • validation rule (if required)

Contraindication Check	We recommend that you define a preliminary questionnaire to establish if any physical measurements or the collection of certain biosamples are contradicated.	You need to define the same information as for any questionnaire. In addition, we need to decide which answers contradindicate which stages.
Physical measurement stages	A physical measurement stage usually acquires one type of measurement on one instrument. However, if several measurements are normally taken together, they can all be grouped in one physical measurement stage. Measurements can be entered manually or automatically captured from electronic instruments (if instrument has its own application, or both. A measurement can be obtained once or multiple times.	Define the list of measurements to be acquired. For each measurement, indicate: • type of measurement • onscreen directions to staff about how to do measurement • data entry mode: manual entry or automatic data capture • for automatic data capture from an instrument, name any inputs that instrument needs to receive from Onyx before it can measure • field(s) for measurements with unit of measure and validation rule for the field • number of times each measurement should be taken
Biosample collection stages	Several types of sample can be acquired. Multiple samples of each type can be acquired. One or more sample collection stages can be defined: one type of sample per stage, or several types per stage—whatever is convenient for the study. Barcode labels on samples can be scanned. The barcode can have a syntax (participant code, sample type, incremental number, and so on).	List of sample collection stages and type(s) of samples to be collected in each stage. For each sample collection stage, indicate: • type(s) of sample • onscreen directions for staff about how to collect samples • will barcodes labels be used? Provide barcode syntax, if known. • how many samples of each type are expected • whether or not you need to check for contraindications • a list of predefined comments that staff could select to describe a sample if necessary
Report(s) to be given to participant	Optional. You can define one or more reports that staff can print and give to participants after completing a stage or at the end of the interview.	Define a list of report(s) that will be printed and given to participants. For each report, pepare the text in a Word document.
Interview Conclusion	Optional. You can define a conclusion stage. For example, a checklist for staff members to go over at the end of the interview.	Define the text to display onscreen

Experimental conditions	Optional. You can define one or more logs in which staff will record experimental conditions. For example, you could have one log for "Room Conditions" (like temperature and relative humidity), and another log for "Location Characteristics" (like meters above sea level).	Define a list of logs. For each log, define: • log name (to be displayed) • a list of condition variables (there may be only one) For each condition, define: • condition name (to be displayed) • type (integer, text, and so on) • unit of measure (if appropriate) • validation rule (if necessary)
Instrument calibration	Optional. You can define calibration procedures for physical measurements. For a particular physical measurement, your staff might need to perform one or several calibrations. For example, for a measurement of "Standing Height", they might need to calibrate "Vertical Alignment" and "Accuracy"	A list of the calibration procedures. For each calibration, indicate: • name of calibration (to be displayed) • name of the physical measurement to which the calibration is related • onscreen directions for staff • list of calibration variables (there may be only one) For each calibration variable, indicate: • type of value (integer, text, and so on) • unit of measure (if appropriate) • validation rule (if necessary)
Data export	Data can be exported to xml files. You can define: • one or more export destinations • the types of data (participant, experimental conditions, instrument calibrations) that you want to export to each destination • which data you want to be export for each type of data • whether or not to encrypt participant data The names of export files will be composed of the destination name and the date and time of export.	Define a list of export destinations For each export destination, define: • destination name (it will be included in the name of the export file) • types of data (participant, experimental conditions, instrument calibrations) to export to this destination For each type of data, define: • the variables to be exported • for participant data, whether or not to encrypt the data • any particularities: for example, timeframe used to select data for export

Who Should Read this Guide

This guide is mainly for **Onyx customizers**. By a *customizer*, we mean a programmer who has been asked by an organization to customize Onyx

to meet the requirements of their research study. Ideally, this person is somewhat or very familiar with Java and how Java implements web applications. The result of the customizer's work will be a war file that will be used to install the custom version of Onyx on a server at the site where Onyx will be used.

This guide may also be used by **Onyx installers** to tweak the Onyx configuration immediately after installation or later. By an *installer*, we mean the person who will set up the Onyx server and workstations at the site where Onyx will be used. An installer may be a system administrator, an IT person, or even the customizer.

Setting Up Your Environment

Overview

To set up your environment so you are ready to prepare a customized version of Onyx, you must do the following:

- Obtaining the Required Software
- Configuring Tomcat
- Configuring MySQL
- Trying the onyx-demo application
    - Troubleshooting
    - Onyx User Guide

Obtaining the Required Software

Software	Suggested Version	Download Link	Use
Java Development Kit	>= 1.8.x	http://java.sun.com/javase/downloads/index.jsp	You need it to generate a war file and if you're going to build custom questionnaires
Tomcat	>= 7.0.x	http://tomcat.apache.org/download-70.cgi	Servlet container - needed to run Onyx
MySQL	>= 5.x	http://www.mysql.com/downloads/mysql/	Database management system
Maven	>= 3.x	http://maven.apache.org/download.html	You will use Maven to build custom questionnaires and to generate the war file that will be used to deploy your version of Onyx.
Acrobat Pro	8.x or 9.x	http://www.adobe.com/products/acrobatpro/tryout.html	You need Acrobat if you are going to create electronic consent forms
onyx-demo project (source files and more)	latest	https://github.com/obiba/onyx-demo/archive/master.zip	We recommend that you use a GIT client to check out (copy) these files, - src (source files including examples of how to write questionnaire classes) - pom.xml (Maven project file which includes all the dependencies you need to build questionnaires and war file) - exampleKeystore.jks (needed to encrypt data upon export) - settings (project settings if you're working in eclipse)

Configuring Tomcat

After installing Tomcat, you must increase the memory allocated to Tomcat's VM as explained in Tips and Tricks for a Successful Installation.

Configuring MySQL

Before starting Onyx for the first time:

1. Create a MySQL database instance using the settings in the table below (execute the command "create database onyx" in a MySQL client).
2. Grant all privileges (CREATE TABLE, ALTER, etc.) to the MySQL user (also "onyx") for that database instance.

Setting	Value
hostname:port	localhost:3306
database name	onyx
username	onyx
password	onyx-demo

Trying the onyx-demo application

We recommend that you familiarize yourself with Onyx by deploying and accessing the onyx-demo application as follows:

1. Get the war for the onyx-demo application. You can either download the onyx-demo.war file from here or generate it from the source files (by running `mvn package` command from wherever you installed the source files for onyx-demo (where the pom.xml file is located).
2. Put the war file in Tomcat's installation directory ($TOMCAT_HOME/webapps). If Tomcat was already running, it should deploy Onyx automatically. If not, start Tomcat and Onyx will be deployed.
3. Access the Onyx demo application by pointing your browser to http://localhost:8080/onyx-demo.
4. When you access the Onyx demo for the first time, you will be prompted to create the administrator user and also to provide some details about the Onyx instance (study name, site name and site ID). See the screen below.

> Take note of the user name and password you entered. You will need them to access onyx-demo.
> To ensure that the demo appointment list loads properly, you must set **Collection Site ID** to **onyx001**.

Troubleshooting

If Onyx does not start, the first thing to look at is Tomcat's console output which is written in the logs directory. On Windows, the file is called stdout_XXX.log. On Debian/Ubuntu, the file is called catalina.out.

Onyx User Guide

FYI, the latest version of the *Onyx User Guide* can be downloaded here.

Customizing the Global Configuration

Overview

To create a customized version of Onyx, you use the onyx-demo application as a starting point. You can modify various aspects of the global Onyx configuration to suit your study's needs, build one or more custom questionnaires, define custom physical measurement and sample collection stages, and then generate a war file so it can be deployed on a server.

- Copying the onyx-demo Project
- Onyx Configuration
    - The WEB-INF/config directory (for customizers)
    - The WEB-INF/config directory (for installers)
    - Onyx Configuration Files
- Configuring the Database Settings
- Configuring Users Authentication and Authorization
    - Configuring Onyx Realm

Copying the onyx-demo Project

The first step in creating a customized version of Onyx is to copy the `onyx-demo` project as and rename it for your study. You will modify the configuration as explained in this guide.

Onyx Configuration

The WEB-INF/config directory (for customizers)

Most of the customization and configuration changes you will make, will be done in the following directory of your project:

        `src/main/webapp/WEB-INF/config`

Throughout this guide, we refer to this directory as your `WEB-INF/config` directory.

The WEB-INF/config directory (for installers)

If you are looking at this guide after installing a custom version of Onyx, your WEB-INF/config directory is under the Tomcat home directory:

        `$TOMCAT_HOME/webapps/custom-onyx/WEB-INF/config`

Onyx Configuration Files

Most Onyx configuration is done in files located in the `WEB-INF/config` directory and its subdirectories.

Most global parameters are set in the `onyx-config.properties` file which is directly in `WEB-INF/config`.

Configuring the Database Settings

The database settings are specified at the end of the `onyx-config.properties` file. If necessary, you can change these settings as indicated in this table.

Property Key	Description
org.obiba.onyx.datasource.driver	the JDBC driver to use (note that the driver's JAR file must be placed in `WEB-INF/lib`)
org.obiba.onyx.datasource.url	the JDBC url for connecting to the database
org.obiba.onyx.datasource.username	the username to use for connecting
org.obiba.onyx.datasource.password	the password to use for connecting
org.obiba.onyx.datasource.dialect	the Hibernate dialect to use (more info here)
org.obiba.onyx.datasource.validationQuery	a valid SQL query to issue for testing JDBC connections
org.obiba.onyx.datasource.testOnBorrow	whether to test pooled connections before providing them

Configuring Users Authentication and Authorization

Onyx integrates Apache Shiro security framework for authenticating and authorizing Onyx users. Shiro has the concept of Realm against which a user can be authenticated and authorization information can be checked. By default Onyx defines its own realm which data are stored in its database.

Different configurations can be applied:

* Onyx realm configuration,
* Adding new realms.

Configuring Onyx Realm

Onyx manages its own database of users. Some configuration can be specified so that Onyx matches your organization requirements in matter of user authentication:

* Password hashing algorithm: the user passwords are not stored as-is in the database, the hashing algorithm can be specified.
* Password validation strategy: when a user is added password complexity constraint can be applied.

The following settings can be edited in the `onyx-config.properties` file:

WEB-INF/config/onyx-config.properties

```
# Algorithm used to hash user passwords.
org.obiba.onyx.password.hash.algorithm=SHA

# Password Validation Strategy
org.obiba.onyx.password.validation.allowedCharacterGroups="A-Z","a-z","0-9
","[.!@#$%&*()]"
org.obiba.onyx.password.validation.minimumCharacterGroupsUsage=3
org.obiba.onyx.password.validation.preventUserAttributeUsage=true
org.obiba.onyx.password.validation.minimumSize=8
org.obiba.onyx.password.validation.maximumSize=14
```

Adding other Realms

The simplest way to add other realms is to edit a `Shiro INI` file (see the one provided with `onyx-demo` project: shiro.ini).

First step is to declare where the `Shiro INI` file can be found in the `onyx-config.properties` file:

WEB-INF/config/onyx-config.properties

```
# Optional Shiro Ini file path to load additional authentication realms
# See http://shiro.apache.org/configuration.html#Configuration-INISections
org.obiba.onyx.authentication.shiro.ini=file:/etc/onyx/config/shiro.ini
```

Then in this INI file specify new realms...

Active Directory Realms

Here is an example for Active Directory:

shiro.ini

```
[main]
# Realms and SecurityManager configuration

# Active Directory realm configuration
# See
http://shiro.apache.org/static/current/apidocs/org/apache/shiro/realm/acti
vedirectory/ActiveDirectoryRealm.html
activeDirectoryRealm =
org.apache.shiro.realm.activedirectory.ActiveDirectoryRealm
activeDirectoryRealm.systemUsername = uid=admin,ou=system
activeDirectoryRealm.systemPassword = secret
activeDirectoryRealm.searchBase = o=sevenSeas,ou=people
activeDirectoryRealm.url = ldap://localhost:10389
# Specify mapping between Active Directory groups and Onyx roles
activeDirectoryRealm.groupRolesMap = group1:SYSTEM_ADMINISTRATOR,
group2:PARTICIPANT_MANAGER, group3:DATA_COLLECTION_OPERATOR
```

LDAP Realms

Here is an example for LDAP (OpenLDAP in this case):

<table>
<tr><td align="center">shiro.ini</td></tr>
</table>

```
[main]
# Realms and SecurityManager configuration

# LDAP realm configuration
ldapRealm = org.obiba.security.realm.LdapRealm
ldapRealm.userDnTemplate = uid={0},ou=people,dc=example,dc=com
ldapRealm.contextFactory.url = ldap://localhost
ldapRealm.contextFactory.authenticationMechanism = none
ldapRealm.contextFactory.systemUsername = admin
ldapRealm.contextFactory.systemPassword = secret
ldapRealm.searchBase = dc=example,dc=com
ldapRealm.userGroupAttribute = memberUid
ldapRealm.groupNameAttribute = cn
# Specify mapping between LDAP groups and Onyx roles
ldapRealm.groupRolesMap = group1:SYSTEM_ADMINISTRATOR,
group2:PARTICIPANT_MANAGER, group3:DATA_COLLECTION_OPERATOR
```

Static user/password

You can also specify statically users with their password and roles.

<table>
<tr><td align="center">shiro.ini</td></tr>
</table>

```
[users]
# The 'users' section is for statically-defined
# set of User accounts.
# Valid Onyx roles are: SYSTEM_ADMINISTRATOR, PARTICIPANT_MANAGER,
PARTICIPANT_RECEPTIONIST, DATA_COLLECTION_OPERATOR
# Format is:
#username=password[,role]*

admin=password,SYSTEM_ADMINISTRATOR
manager=password,PARTICIPANT_MANAGER
reception=password,PARTICIPANT_RECEPTIONIST
operator1=password,DATA_COLLECTION_OPERATOR
operator2=password,DATA_COLLECTION_OPERATOR
```

Finally, in case of an added realm does not return user roles that are relevant to Onyx, a roles map can be specified in the `onyx-config.prope rties` file:

<table>
<tr><td align="center">WEB-INF/config/onyx-config.properties</td></tr>
</table>

```
# Define mapping between roles from other realms to Onyx roles
# Valid Onyx roles are: SYSTEM_ADMINISTRATOR, PARTICIPANT_MANAGER,
PARTICIPANT_RECEPTIONIST, DATA_COLLECTION_OPERATOR
org.obiba.onyx.authentication.rolesMap=role1=DATA_COLLECTION_OPERATOR,role
2=DATA_COLLECTION_OPERATOR,role3=PARTICIPANT_MANAGER,role4=PARTICIPANT_REC
EPTIONIST
```

Configuring Opal Connection

You would need to configure connection with a remote Opal server for:

- Connecting a datasource from a remote Opal server
- Exporting Data directly to Opal

WEB-INF/config/onyx-config.properties

```
[...]
# Opal URL and credentials
org.obiba.onyx.opal.url=http://localhost:8081/
org.obiba.onyx.opal.username=administrator
org.obiba.onyx.opal.password=password
# Opal connection advanced settings
# Timeout in milliseconds until a connection is established. A timeout
value of zero is interpreted as
# an infinite timeout. Default value is 10000.
# org.obiba.onyx.opal.connectionTimeout=10000
# Socket timeout in milliseconds, which is the timeout for waiting for data
or, put differently, a maximum
# period inactivity between two consecutive data packets). A timeout value
of zero is interpreted as an
# infinite timeout. Default value is 600000.
# org.obiba.onyx.opal.soTimeout=600000
[...]
```

Configuring Participant Recuitment

Onyx supports two ways of recruiting participants: ENROLLED (participants who have appointments) and VOLUNTEER (participants without appointments, also known as "walk-ins"). Onyx can be configured to accept either recruitment type or both. This is set near the top of the `onyx-config.properties` file.

WEB-INF/config/onyx-config.properties

```
[...]
# A comma separated list of supported recruitment types
org.obiba.onyx.supportedRecruitmentTypes=ENROLLED,VOLUNTEER
[...]
```

Configuring Participant Attributes

Onyx collects data about people who agree to participate in a study. These people are called "participants".

Built-in participant attributes

Onyx includes the following attributes for all participants. These attributes can be customized as explained below:

Attribute	Description
Enrollment ID	An identifier for the participant that was assigned at enrollment time. Typically, this value is read from the appointment list.

Participant ID	The unique identifier used in Onyx. You can define a format for the Participant ID as explained below.
First Name	
Last Name	
Birth Date	
Gender	Possible values are: M,F
Assessment Center ID	Also
Appointment Time	

Defining the format of the Participant ID (optional)

You can define a validation pattern for the Participant ID attribute in the following section of the `onyx-config.properties` file.

WEB-INF/config/onyx-config.properties

```
[...]
# Participant Id validation pattern
org.obiba.onyx.participantId.pattern=.*
[...]
```

The default pattern (.*) means any value would be accepted. For detailed information about the defining a validation pattern, see JavaPattern.

Customizing participant attributes

Participant attributes can be customized in the `participant-attributes.xml` file in `WEB-INF/config`. In this file, you can modify existing attributes and declare new ones. Modifying built-in attributes is explained below.

A new attribute can be declared using the following schema:

Attribute	Possible Values	Description
name		Localizable attribute name
assignableAtEnrollment	true, false	true means this attribute must be present in the appointment list. A value may be set at enrollment time, but it is not required.
mandatoryAtEnrollment	true, false	true means a value must be set for this attribute at enrollment time. If this is set to true and there is no value when the appointment is being updated, an error will resut. If this is set to false, this property does not need to be set at enrollment time.
mandatoryAtReception	true, false	true means this attribute must be set when the participant is received through the Onyx user interface.
editableAtReception	true, false	true means the value of this attribute can be changed when the participant is received through the Onyx user interface.

`editableAfterReception`	`true, false`	`true` means the value of this attribute can be changed at any time during the participant interview.
`type`	TEXT, INTEGER, DECIMAL, BOOLEAN, DATE	The type of the attribute value
`allowedValues`	list of strings	Localizable allowed values for the attribute
`validators`	patternValidator, rangeValidator	Validation rules of the attribute value

For example, these are the custom attributes of participants defined for the onyx-demo application:

WEB-INF/config/participant-attributes.xml

```xml
<participantAttributes>
  <attribute>
    <name>Country</name>
    <assignableAtEnrollment>true</assignableAtEnrollment>
    <mandatoryAtEnrollment>false</mandatoryAtEnrollment>
    <mandatoryAtReception>false</mandatoryAtReception>
    <editableAtReception>true</editableAtReception>
    <editableAfterReception>true</editableAfterReception>
    <type>TEXT</type>
    <allowedValues>
      <string>CA</string>
      <string>US</string>
    </allowedValues>
  </attribute>
  <attribute>
    <name>Postal Code</name>
    <assignableAtEnrollment>true</assignableAtEnrollment>
    <mandatoryAtEnrollment>false</mandatoryAtEnrollment>
    <mandatoryAtReception>false</mandatoryAtReception>
    <editableAtReception>true</editableAtReception>
    <editableAfterReception>true</editableAfterReception>
    <type>TEXT</type>
    <validators>
      <patternValidator>^[ABCEGHJKLMNPRSTVXYabceghjklmnprstvxy]
        {1}\d{1}[A-Za-z]{1}
        *\d{1}[A-Za-z]{1}\d{1}$</patternValidator>
    </validators>
  </attribute>
  <attribute>
    <name>Phone</name>
    <assignableAtEnrollment>true</assignableAtEnrollment>
    <mandatoryAtEnrollment>false</mandatoryAtEnrollment>
    <mandatoryAtReception>false</mandatoryAtReception>
    <editableAtReception>true</editableAtReception>
    <editableAfterReception>true</editableAfterReception>
    <type>TEXT</type>
    <validators>
      <rangeValidator type="TEXT">
        <minimum>1</minimum>
        <maximum>50</maximum>
      </rangeValidator>
    </validators>
  </attribute>
</participantAttributes>
```

To localize the names of participant attributes

The names of participant attributes can be localized in the{{ messages_*.properties }}files in }}{{ WEB-INF/config. For onyx-demo, English and French properties files are provided.

To validate participant attributes

You can define patternValidator which will validate the value entered based on a regular expression as specified in JavaPattern.

The range validator is able to check that the participant value is in a range of numerical values (INTEGER, DECIMAL), dates (DATE) or text length (TEXT). Upper and lower bound can be provided individually or both.

Allowed date formats for date bounds are:

- `yyyy-MM-dd`
- `yyyy-MM-dd HH:mm:ss.S`

Here are some examples of how to configure range validators:

```
<rangeValidator type="INTEGER">
   <minimum>10</minimum>
</rangeValidator>


<rangeValidator type="DECIMAL">
   <maximum>1.5</maximum>
</rangeValidator>


<rangeValidator type="DATE">
   <minimum>1099-01-01</maximum>
   <maximum>2099-01-01</maximum>
</rangeValidator>


<rangeValidator type="TEXT">
   <minimum>10</maximum>
   <maximum>250</maximum>
</rangeValidator>
```

Customizing essential participant attributes

To modify the essential participant attributes the file `essential-participant-attributes.xml` must be present in the `WEB-INF/config` directory. If `essential-participant-attributes.xml` is not part of the configuration there are two methods for obtaining it.

1. Download `essential-participant-attributes.xml` from this page. The essential attributes are stable and not regularly modified so downloading them is relatively safe.
2. Extract the file from Onyx. This option is more complicated, but will guarantee that the correct `essential-participant-attributes.xml` file is used. The steps to extract the file are detailed below.

In order to modify the essential Participant attributes, first extract the `essential-participant-attributes.xml` file from the `onyx-core*.jar` file and add it to the existing cohort configuration alongside the existing `participant-attributes.xml` file.

```
# Make your current directory the one containing the Onyx web application
$ mkdir temp
$ cp WEB-INF/lib/onyx-core-*.jar temp
$ cd temp
$ jar -xvf onyx-core-*.jar
$ cp META-INF/essential-participant-attributes.xml
<cohort-configuration-root-directory>/src/main/webapp/WEB-INF/config/
```

Once the `essential-participant-attributes.xml` file has been made part of the configuration it may be modified. It is not possible to add new essential attributes or remove essential attributes, but it is possible to modify the definition of these attributes in various ways. For example, attributes that are not editable at the time of reception may be made so by modifying the value of <editableAtReception> from false to true.

Configuring the Variables

Defining Custom Variables

You can define custom variables in a file called `custom-variables.xml` file in `WEB-INF/config`. Here are some reasons you might want to define a custom variable:

- for export purposes (see the variable `PostalCodePrefix` in the example below)
- to use in a condition when defining stage dependencies (see Defining Stage Dependencies)
- to use in a condition used to determine whether or not to display a particular question in a questionnaire

For information on how to define a `script`, see Magma Javascript API.

WEB-INF/config/custom-variables.xml

```xml
[...]
<variables>

 <variable name="Admin.Participant.PostalCodePrefix" valueType="text"
entityType="Participant">
  <attributes>
   <attribute name="label" valueType="text" locale="en">Postal Code
Prefix</attribute>
   <attribute name="label" valueType="text" locale="fr">Préfixe du code
postal</attribute>
   <attribute name="script" valueType="text">
$('Admin.Participant.Postal
Code').replace(/^\s*([a-zA-Z|a-zA-Z]\d[a-zA-Z|a-zA-Z]).*$/, '$1')
   </attribute>
  </attributes>
 </variable>

 <variable name="Declared_Gender" valueType="text"
entityType="Participant">
   <attributes>
     <attribute name="label" valueType="text" locale="en">Declared
Gender</attribute>
     <attribute name="label" valueType="text" locale="fr">Genre
déclaré</attribute>
     <attribute name="script" valueType="text">
        $('IdentificationQuestionnaire:Gender')
     </attribute>
   </attributes>
   <categories>
     <category name="MALE">
       <attributes>
         <attribute name="label" valueType="text"
locale="en">Male</attribute>
         <attribute name="label" valueType="text"
locale="fr">Homme</attribute>
       </attributes>
     </category>
     <category name="FEMALE">
       <attributes>
         <attribute name="label" valueType="text"
locale="en">Female</attribute>
         <attribute name="label" valueType="text"
locale="fr">Femme</attribute>
       </attributes>
     </category>
   </categories>
 </variable>

</variables>
```

Connecting a datasource

Onyx can access any of the collected participant data through the Magma Javascript API.

Other datasources can be connected to Onyx in order to make additional information available to Onyx. This information can be about the Participants or about other entities (Drugs, Geographical Area etc.).

Caching can be activated for some type of datasources: the caching feature allows to speed up data extraction from the datasource (file or remote server based); it also allows to run onyx in disconnected mode (cache can be persisted on disk). See more about caching in the Configuring the Cache section.

Currently the following types of datasource are supported:

Type	Description	Caching supported
CSV	Connection to a CSV file	✅
XML	Connection to a Opal-XML file	❌
Opal	Connection to a remote Opal server	✅

All these datasource types can be combined together in `WEB-INF/spring/custom.xml` file.

Connecting a CSV datasource

The following example declares a CSV file containing medications information (ID, label, manufacturer, indications etc.).

WEB-INF/spring/custom.xml

```xml
<beans xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-2.5.xsd">

   <bean class="org.obiba.onyx.opal.CsvDatasourceProvider"
lazy-init="false">
     <property name="magmaEngine" ref="magmaEngine" />

     <!-- Path to the CSV file -->
     <property name="file"
value="${org.obiba.onyx.config.path}/datasources/medications/Drugs.csv" />

     <!-- Name of the datasource that will be used in Onyx scripts -->
     <property name="datasourceName" value="medications" />

     <!-- Name of the table that will be used in Onyx scripts -->
     <property name="tableName" value="Drugs" />

     <!-- Type of the entity (optional, default is Participant) -->
     <property name="entityType" value="Drug" />

     <!-- Caching activation (default is false) -->
     <property name="withCaching" value="true"/>
   </bean>

</beans>
```

Connecting a XML datasource

The following example declares a Opal-XML datasource, by providing a Opal-XML archive (zip file). This kind of

WEB-INF/spring/custom.xml

```
<beans xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-2.5.xsd">

   <bean class="org.obiba.onyx.opal.FsDatasourceProvider" lazy-init="false">
     <property name="magmaEngine" ref="magmaEngine" />

     <!-- Path to the XML file, declared in
WEB-INF/config/onyx-config.properties in my.path.to.the.xml.archive.file
property -->
     <property name="file" value="${my.path.to.the.xml.archive.file}" />

     <!-- Name of the datasource that will be used in Onyx scripts -->
     <property name="datasourceName" value="inhome-interview" />

   </bean>

</beans>
```

Connecting a datasource from a remote Opal server

The Opal connection properties must be defined in `WEB-INF/config/onyx-config.properties` using `org.obiba.onyx.opal.*` properties.

WEB-INF/spring/custom.xml

```xml
<beans xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-2.5.xsd">

   <bean class="org.obiba.onyx.opal.OpalDatasourceProvider"
lazy-init="false">
     <property name="participantService" ref="participantService" />
     <property name="magmaEngine" ref="magmaEngine" />
     <property name="opalJavaClient" ref="opalJavaClient" />

     <!-- Name of the datasource in Opal -->
     <property name="opalDatasourceName" value="opal-data"/>

     <!-- Name of the datasource that will be used in Onyx scripts -->
     <property name="onyxDatasourceName" value="inhome-interview"/>

     <!-- Caching activation (default is false) -->
     <property name="withCaching" value="true"/>

   </bean>

</beans>
```

Configuring the Cache

Onyx uses EhCache for performing tasks related to caching. If you are interested in activating caching of external datasources, you will have to override the default cache configuration file with your settings: `WEB-INF/config/ehcache-onyx.xml`.

This is an example of configuration that matches the Connecting a datasource examples:

WEB-INF/config/ehcache-onyx.xml

```xml
<ehcache xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="http://ehcache.org/ehcache.xsd"
updateCheck="false">

  <diskStore path="/path/to/cache/folder"/>

  <defaultCache
      maxElementsInMemory="10000"
      eternal="false"
      timeToIdleSeconds="120"
      timeToLiveSeconds="120"
      overflowToDisk="true"
      diskPersistent="false"/>

  <!-- Cache name pattern: datasource-[name of the datasource in onyx] -->

  <cache name="datasource-inhome-interview"
       maxElementsInMemory="10"
       eternal="true"
       timeToLiveSeconds="0"
       timeToIdleSeconds="0"
       overflowToDisk="true"
       diskPersistent="true"/>

  <cache name="datasource-medications"
      maxElementsInMemory="10000"
      eternal="true"
      timeToLiveSeconds="0"
      timeToIdleSeconds="0"
      overflowToDisk="true"
      diskPersistent="true"/>
</ehcache>
```

Configuring the Appointment list

Onyx imports participant appointments from a file (generated by some other programme). The file can be in Excel or XML format.

Defining input and output directories for the appointment list

You must define the input directory where Onyx will look for the appointment list file. Defining an output directory is optional. When updating the appointment list, the user can select a file. If they don't, Onyx will process whatever XML or Excel (.XSL) file it finds in the input directory.

<table>
<tr><th colspan="1">WEB-INF/config/onyx-config.properties</th></tr>
<tr><td>

```
# Directory that contains the appointment list files
# (dropped by external process)to be processed
org.obiba.onyx.appointments.inputDirectory=WEB-INF/appointments/in
# Optional directory that contains the successfully processed files
org.obiba.onyx.appointments.outputDirectory=WEB-INF/appointments/out
```
</td></tr>
</table>

Configuring the appointment list reader

The appointment list file is read by a `ParticipantReader` class. You set a number of lines in the `onyx-config.properties` file to define what the reader class should look for in the appointment file. See the table below for an explanation of the attributes related to the participant reader. See also the Excel example and the XML example.

Attribute	Needed for	Description
org.obiba.onyx.participantReader.sheetNumber	Excel appointment lists	Excel workbooks can have multiple worksheets. This attribute identifies which worksheet to read.
org.obiba.onyx.participantReader.headerRowNumber	Excel appointment lists	This indicates which row contains the names of columns. The column names are mapped to participant attributes (in the columnToAttribute attribute which is explained below in this table).
org.obiba.onyx.participantReader.firstDataRowNumber	Excel appointment lists	Indicates the first row in the sheet that contains participant data.
org.obiba.onyx.participantReader.columnToAttribute	Excel and XML appointment lists	*For Excel format appointment lists:* Maps the name of columns in the Excel file onto Onyx participant attributes. In the example below, the column labelled "Sex" is mapped onto the participant attribute "Gender". If the name of the column and the participant attribute are the same, they don't need to be included in the mapping. *For XML format appointment lists:* Maps the name of elements in the XML file onto Onyx participant attributes. In the example below, the attribute with the key "Sexe" is mapped onto the participant attribute "Gender". If the name of the column and the participant attribute are the same, they don't need to be included in the mapping.

Example Excel appointment list

For an Excel format appointment list that looks like this (click on the thumbnail below)...

...the participant reader attributes would be set as follows in `onyx-config.properties`:

```
[...]
# Participant reader
org.obiba.onyx.participantReader.sheetNumber=1
org.obiba.onyx.participantReader.headerRowNumber=1
org.obiba.onyx.participantReader.firstDataRowNumber=2
# List of key/value pairs separated by a comma. For example,
"param1=foo,param2=bar".
org.obiba.onyx.participantReader.columnToAttribute=Participant
ID=Enrollment ID,
    Collection Site Id=Assessment Center ID,Sex=Gender,Appointment
date/time=Appointment Time
[...]
```

Example XML appointment list

For an XML format appointment list that looks like this...

```
[...]

<participants>
  <participant>
    <attribute key="Code_participant" value="404"/>
    <attribute key="Site" value="onyx001"/>
    <attribute key="Sexe" value="M"/>
    <attribute key="Date_heure_RDV" value="2009-10-29 03:30:00"/>
    <attribute key="Prenom" value="Harry"/>
    <attribute key="Nom" value="Belafonte"/>
    <attribute key="Date_naissance" value="1928-10-01 00:00:00"/>
  </participant>

[...]
```

...the participant reader attributes would be set as follows in `onyx-config.properties`:

WEB-INF/config/onyx-config.properties (XML format appointment list)

```
\# Participant reader
\# List of key/value pairs separated by a comma. For example,
"param1=foo,param2=bar".
org.obiba.onyx.participantReader.columnToAttribute=Code_participant=Enroll
ment ID,
    Site=Assessment Center ID,Sexe=Gender,Date_heure_RDV=Appointment Time,
    Prenom=First Name,Nom=Last Name,Date_naissance=Birth Date
```

Scheduling automatic updates of the appointment list

Edit the following line in the `onyx-config.properties` file:

```
# Schedule for automatic appointment list updates
org.obiba.onyx.appointments.schedule=0 0 4 * * ?
```

The above configuration ("0 0 4 * * ?") means that the appointment list will be updated every day at 4:00 a.m. To disable automatic updates, simply leave the property blank, as follows:

```
org.obiba.onyx.appointments.schedule=
```

For details of the syntax to use, see http://quartz.sourceforge.net/javadoc/org/quartz/CronTrigger.html.

Cache Warm-up

If some external datasource caching has been configured, it is possible to warm-up the cache: fetch the participants data from the external datasource at appointment list update and optionally make them available in memory cache at participant reception time or when the participant's interview starts.

WEB-INF/spring/custom.xml

```xml
<beans xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-2.5.xsd">

   <!-- ... -->

   <!-- This processor will cache all participant's data (from datasources
having cache activated) after the appointment list has been updated -->
   <bean
class="org.obiba.onyx.core.etl.participant.impl.CachedDatasourceProcessor">
      <property name="magmaEngine" ref="magmaEngine" />

      <!-- Tables to be cached can be explicitly named (in their onyx
datasource) in a comma separated list -->
      <!-- Otherwise participant data from all tables with cache support will
be fetched -->
      <!--<property name="tableNames" value="opal-data.InHome" />-->

      <!-- Cache warm-up at participant reception (default is false) -->
      <property name="proccessOnReception" value="true" />

      <!-- Cache warm-up when participant is interviewed (default is false)
-->
      <property name="proccessOnInterview" value="true" />

   </bean>
</beans>
```

Modifying Date and Time Formats

You can edit the data and time formats that will be used throughout Onyx by editing these lines in the `onyx-config.properties` file:

WEB-INF/config/onyx-config.properties

```
# A default value for a reusable date format
org.obiba.onyx.dateFormat=yyyy-MM-dd
# A default value for a reusable date/time format
org.obiba.onyx.dateTimeFormat=yyyy-MM-dd HH:mm
```

Modifying the Printer

On startup, Onyx decides which printer to use for printing reports for participants. Onyx will try to find a printer with a particular name. If that printer does not exist, or does not support PS printing, then it will fallback to using the system's default printer. If that printer does not support PS printing, then printing reports from Onyx will not be possible.

You can change the printer to look up at startup:

Property Key	Description
org.obiba.onyx.pdfPrinterName	Name of printer to try before trying the system default printer

Configuring Participant ID Generator

By default, Onyx does not generate participant identifiers. It is the Onyx operator that assigns a unique identifier to each participant manually when receiving them at the collection site. In this case, your organization defines the rules for generating these identifiers and Onyx will not be aware of them.

However, if you prefer to let Onyx takes care of this matter, it is possible to configure it so that it generates identifiers based on a pre-configured set of rules:

Property Key	Expected Value	Description
org.obiba.onyx.participantId.generated	true or false	Configures Onyx to generate participant identifiers
org.obiba.onyx.participantId.maxIncrement	integer	The maximum value of the random increment

Based on the configuration described above, Onyx will generate a random number between 1 and maxIncrement which will be added to the last generated identifier sequence. The first identifier (the starting point for all this) and the Identifier Prefix (a list of characters be added before the random number) are specified when Deploying Onyx.

Example configuration

With the following configuration:

Configuration	Value
First Identifier	2
Identifier Prefix	DEMO
maxIncrement	5

The resulting identifiers should look like this:

DEMO + 2 (first identifier) + 3 (generated increment) = **DEMO5**
DEMO + 5 (previous identifier) + 4 (generated increment) = **DEMO9**
DEMO + 9 (previous identifier) + 2 (generated increment) = **DEMO11**
and so on...

Configuring the Participant Registry

The Participant Registry makes it possible to enroll new participants by retrieving participant information from an external system via a RESTful web service call. The Participant Registry is off by default and can be turned on and configured using the following properties.

Property Key	Values	Description
org.obiba.onyx.participant.registry.enabled	true, false	Displays the "Participant Registry" button in the Onyx user interface when true.
org.obiba.onyx.participant.registry.type	restful, fixed	The restful participant registry will use the url below to retrieve participants from an external service. The fixed participant registry is used for testing.
org.obiba.onyx.participant.registry.uriTemplate	url (e.g. http://localhost/participant/{identifier})	This is the url used by the participant registry to retrieve participants. The {identifier} string will be replaced by the string entered in by the user in the participant registry search field.

The Fixed Participant Registry

Configuring the fixed participant registry will provide a demonstration of the Participant Registry User Interface by returning canned participants for known participant id values. Developers also use this registry when testing and developing the Participant Registry user interface. The behaviour of the fixed participant registry is described below.

Id provided for lookup	Result
1 - 4	Returns a canned Participant
5	Behaves as if the Participant Registry failed
anything else	Behaves as if the participant was not found

The RESTful Participant Registry

The RESTful participant registry expects a specific response from a third party web service. It is up to third parties to build the RESTful web service that will supply Onyx with participant information. Here are the specific details about what can be included in the response.

- The expected content type of the response is `text/xml`.
- The service should send the HTTP code 200 (ok) when returning participant XML.
- To indicate that a participant was not found the service should return HTTP code 404 (not found).
- The Restful Participant Registry will interpret any other HTTP code sent as a participant registry lookup failure.

The expected body of the response is XML representing a single Participant. The example below shows the names of the XML elements Onyx requires in order to de-serialize the participant.

```
<org.obiba.onyx.core.domain.participant.Participant>
  <firstName>Bodvild</firstName>
  <lastName>Nidudsdatter</lastName>
  <gender>FEMALE</gender>
  <birthDate>1960-01-23 11:59:51.703 EST</birthDate>
  <recruitmentType>VOLUNTEER</recruitmentType>
  <configuredAttributeValues>
    <org.obiba.onyx.core.domain.participant.ParticipantAttributeValue>
      <attributeName>Street</attributeName>
      <attributeType>TEXT</attributeType>
      <textValue>392 Iduns Gate</textValue>
      <participant reference="../../.."/>
    </org.obiba.onyx.core.domain.participant.ParticipantAttributeValue>
    <org.obiba.onyx.core.domain.participant.ParticipantAttributeValue>
      <attributeName>City</attributeName>
      <attributeType>TEXT</attributeType>
      <textValue>Oslo</textValue>
      <participant reference="../../.."/>
    </org.obiba.onyx.core.domain.participant.ParticipantAttributeValue>
  </configuredAttributeValues>
</org.obiba.onyx.core.domain.participant.Participant>
```

Configuring Interview Stages

Overview

An Onyx interview consists of stages. Four types of stage are possible: consent, physical measurment, sample collection, questionnaire.

- Configuration Files for Interview Stages
- Preparing the Stages
- Defining Stage Dependencies
  - To view a list of Onyx variables
- Defining a Conclusion Stage
- Ordering the Stages
- Configuring the Action Windows for Stages
  - Naming and invoking of action windows
  - Suppressing action windows
  - Fine-tuning action windows
    - Customizing the text above the comment field

Configuration Files for Interview Stages

Onyx stages are configured in different subdirectories of the `WEB-INF/config` directory. Each type of stage has its own subdirectory:

- The Consent Stage is configured in: `WEB-INF/config/marble`
- Physical Measurement Stages are configured in: `WEB-INF/config/jade`
- Sample Collection Stages are configured in: `WEB-INF/config/ruby`
- Questionnaire Stages are configured in: `WEB-INF/config/quartz`

Each of these subdirectories, contains the following files:

- `module-context.xml` (contains context information for the Spring framework)
- `module-config.properties` (contains configuration properties used by Onyx)
- `messages_locale.properties` (contains localized strings for onscreen labels)
- `stages.xml` (contains stage definitions)

Preparing the Stages

You prepare the stages required for your custom version of Onyx as explained in these pages:

- Customizing the Consent Stage
- Building Custom Questionnaires
- Configuring Physical Measurement Stages
- Configuring Sample Collection Stages

Defining Stage Dependencies

When you add a stage to Onyx (by adding a `<stage>` element to the appropriate `stages.xml` file), you can define dependencies for the stage. In other words, you can set conditions that will be evaluated at runtime to determine the state of the stage (READY or some other state).

A stage dependency condition is composed of two different statements:

- a list of stages that are expected to be in a final state (i.e. completed or skipped) before any condition is evaluated,
- a script using Magma javascript API that evaluates the participant's data referred by variables. The result of the script is a boolean (true/false value).

For example: The BloodPressure stage in onyx-demo, depends on two conditions: (1) the participant accepting the consent form and (2) completion of the CIPreliminaryQuestionnaire indicating that the Blood Pressure is not contraindicated.

WEB-INF/config/jade/stages.xml

```
[...]
 <stage>
    <name>BloodPressure</name>
    <module>jade</module>
    <stageDependencyCondition class="scriptCondition">
      <stages>
        <string>Consent</string>
        <string>CIPreliminaryQuestionnaire</string>
      </stages>
      <script>
        $('Consent:accepted')
        .and($('CIPreliminaryQuestionnaire:BP_CI').not())
      </script>
    </stageDependencyCondition>
  </stage>

[...]

</stages>
```

To view a list of Onyx variables

To view a list of the variables that you can use in stage dependency conditions, you must run Onyx in development mode and use a utility to output a zip file containing variable lists and other information.

1. Edit `onyx-config.properties` and set the application mode to development as follows:

WEB-INF/config/onyx-config.properties

```
org.obiba.onyx.webapp.configurationType=development
```

2. Restart Tomcat.
3. Access onyx-demo by pointing your browser to http://localhost:8080/onyx-demo.
4. Enter adminstrator user name and password.

5. Select the **Administration** link in the upper right corner.
6. Select the **Developers** tab.
7. Select data & dictionary format (Excel, XML).
8. Click the **Download** button.

A file named `magma-dump.zip` will be written at the location indicated onscreen. The file `Participants/variables.xml` (in the zip file) lists the variables that are most useful for defining stage dependency conditions.

Defining a Conclusion Stage

Onyx requires a conclusion stage so that an interview can conclude normally and be assigned the "complete" status.

You can configure any stage to be the conclusion stage by editing the `stages.xml` file that contains the stage definition and adding an `<interv iewConclusion>` element to the appropriate `<stage>` element. In the onyx-demo application, the conclusion stage is the ConclusionQuestionnaire which is defined in `WEB-INF/config/quartz/stages.xml`:

```
<stage>
  <name>ConclusionQuestionnaire</name>
  <module>quartz</module>
  <interviewConclusion>true</interviewConclusion>
</stage>
```

If you configure one of your custom stages to be the conclusion stage, be sure to remove the above stage element from the `WEB-INF/config/q uartz/stages.xml` in your project.

Ordering the Stages

In the procedures for preparing the different types of stages, the last step was inserting in the stage ordering in the `onyx-config.properties` f ile as shown below. After you have prepared all the stages for your custom version of Onyx, you should probably check the order.

<div>

WEB-INF/config/onyx-config.properties

```
# A comma separated list of stage names that defines the ordering within an
Interview.
org.obiba.onyx.engine.staticStageOrder=Consent,HealthQuestionnaire,BloodPr
essure,...,Conclusion
```

</div>

The stage names that must be used in this line are those defined in the various `stages.xml` files located in these directories:

- `Consent stage: WEB-INF/config/marble/stages.xml`
- `Questionnaire stages: WEB-INF/config/quartz/stages.xml`
- `Physical measurement stages: WEB-INF/config/jade/stages.xml`
- `Sample Collection stages: WEB-INF/config/ruby/stages.xml`

If no stage can be identified by a provided name, the name will be ignored.

If a stage is defined in a `stages.xml` file, but not included in the stage order list, Onyx will include the stage in the interview.

Configuring the Action Windows for Stages

By default, the onyx-demo application, displays an **action window** whenever the operator performs an action related to a stage. For example, when the operator starts or finishes a stage. These windows are configured in the `WEB-INF/config/action-definitions.xml` file.

The following `actionDefinition` element shows how a typical action window is defined. For information about the elements within an `<actio nDefinition>`, see Fine-tuning action windows.

<table>
<tr><td colspan="1" align="center">**WEB-INF/config/action-definitions.xml**</td></tr>
</table>

```
<actionDefinition>
    <code>action.STOP.InProgress.Marble</code>
    <type>STOP</type>
    <askParticipantId>true</askParticipantId>
    <askPassword>true</askPassword>
    <commentMandatory>true</commentMandatory>
    <reasonMandatory>true</reasonMandatory>
    <reasons>
      <string>Action.Reason.TechnicalProblem</string>
      <string>Action.Reason.ParticipantDecision</string>
      <string>Action.Reason.InterviewerDecision</string>
      <string>Action.Reason.SeeComment</string>
    </reasons>
</actionDefinition>
```

Naming and invoking of action windows

The name or `<code>` of an action window is composed as follows:

```
action.ACTION_TYPE.StageState.StageType
```

where,

- *ACTION_TYPE* is the actual action. Possible values: `EXECUTE`, `STOP`, `SKIP`, `COMPLETE`.

- *StageState* is the state of the stage on which the action was invoked. Possible values: `Ready`, `InProgress`, `Interrupted`, `Completed`.

- *StageType* is the type of stage. Possible values:
  - `Marble` for the consent stage
  - `Jade` for physical measurement stages
  - `Quartz` for questionnaire stages
  - `Ruby` for sample collection stages

When a user takes an action in the Onyx user interface, Onyx determines which action window to display (or not) by looking for the most specific action definition that fits the action, and works its way to the least specific action definition that fits. For example, if Onyx can't find `action.EXECUTE.Stop.Marble`, it might end up using the `action.EXECUTE.Stop` definition.

Suppressing action windows

You cannot suppress the display of all action windows by making a single change in the `action-definitions.xml` file. You must suppress them one-by-one.

To suppress the display of a particular action window, add the following element to the appropriate `<actionDefinition>` element.

```
<askComment>false</askComment>
```

Notes

- If you want to suppress a "start" window, look for actions with this *ACTION_TYPE.StageState* combination: `EXECUTE.Ready`
- The two general `<actionDefinition>` elements at the top of the `action-definitions.xml` file, do not override the more particular `<actionDefinitions>` elements lower down in the file. Adding `<askComment>false</askComment>` to either of these two elements will not have the effect of removing action windows for all stages.

Fine-tuning action windows

The following table explains the effect of setting key elements in the `action-definitions.xml` file in `WEB-INF/config`.

Element	Default value	Description
`<askParticipantId>`	`true`	If `true`, the Participant ID field will be available, and the user must enter the correct Participant ID. If `false`, the Participant ID field will be grayed out.
`<askPassword>`	`true`	If `true`, the Password field will be available, and the user must enter their password. If `false`, the Password ID field will be grayed out.
`<commentMandatory>`	`true`	If `true`, the Comment field will be available, and the user must enter a comment. If `false`, the Comment field will be grayed out.
`<reasonMandatory>`	`false`	If `true`, the Reason drop-down list will be available and populated with the Strings in the `<reasons>` element; and the user must select a reason. If `false`, the Reason drop-down list will be grayed out.
`<askComment>`	`true`	If `true`, an Action window will be displayed. If `false`, an Action window will not be displayed. NOTE: For backward compatibility, if an `<askComment>` element is not present, it is equivalent to setting it to true (so the Action window will be displayed).

Customizing the text above the comment field

To customize the text displayed above the comment field in an action window, you add a <commentNote> element to the appropriate <actionDefinition> element in the action-definitions.xml file.

Here is an example :

```
WEB-INF/config/action-definitions.xml
<actionDefinition>
<code>action.COMPLETE.InProgress.Jade</code>
<type>COMPLETE</type>
<askParticipantId>true</askParticipantId>
<askPassword>false</askPassword>
<commentMandatory>false</commentMandatory>
<reasonMandatory>false</reasonMandatory>
<commentNote>action.COMPLETE.InProgress.Jade.CommentNote</commentNote>
</actionDefinition>
```

Then you need to add localized strings containing the desired message in each of the message-locale.properties files in the WEB-INF/config directory. For example, for the above <commentNote>, you would add the following line to the English properties file:

WEB-INF/config/messages_en.properties
action.COMPLETE.InProgress.Jade.CommentNote=Please enter your comments below.

Each <actionDefinition> element can include its own <commentNote> element. If the configuration doesn't specify a custom message for an action, a default message will be displayed.

Customizing the Consent Stage

Overview

The onyx-demo application that you are using as a basis for your customization, includes a consent stage. More generally, an interview can have multiple consent stages. You can modify the configuration of the consent stages for your customized version of Onyx in these ways:

Configuration Files for the Consent Stage

The configuration files for the consent stage are found in the `WEB-INF/config/marble` directory.

The consent stage is included in Onyx by means of a `<stage>` element in the `stages.xml` file in this directory. The name of the consent stage is defined in this file. In the onyx-demo application, the name of the consent stage is `Consent`.

Configuring Consent Modes (electronic and paper)

The consent stage in the onyx-demo application is configured to accept the participant's consent in electronic and paper forms. Paper mode is always possible (it cannot be disabled). Electronic mode can be disabled.

If your organization will obtain consent electronically, you will need create custom consent forms as explained in Creating Custom Electronic Consent Forms (PDF).

If your organization will only obtain consent on paper forms, you must disable electronic consent by setting the `allowElectronicMode` property to `false` in the `marble/module-config.properties` file as follows:

WEB-INF/config/marble/module-config.properties

```
org.obiba.onyx.marble.consent.allowElectronicMode=false
```

Configuring the Locales for Paper Consent

Your organization may provide its paper consent forms in several languages. The Onyx consent stage allows the user to select the language of the paper consent form given to the participant. The list of available languages is confugured by editing the following line in the `WEB-INF/config/marble/module-config.properties` file.

WEB-INF/config/marble/module-config.properties

```
org.obiba.onyx.marble.consent.manual.availableLocale=en,fr
```

As shown above, the onyx-demo application accepts paper consent in English and French. If your organization requires other languages, add the appropriate two-letter language codes to this line. For possible language codes, see http://www.loc.gov/standards/iso639-2/php/code_list.php.

Creating Custom Electronic Consent Forms (PDF)

If your custom version of Onyx will accept consent electronically, you must create a consent form in each language that is required for your version of Onyx.

> This procedure requires a single license for Acrobat Professional. Onyx uses Acrobat Reader (which is free) to display the consent form at runtime.

To create a PDF consent form

Repeat this procedure to create a PDF consent form for each language that you need to support.

1. Create a consent form in one of your supported languages, using Microsoft Word or any tool of your choice.
2. Convert the form to PDF using a PDF converter.

3. Save the PDF using a name that includes the appropriate language code (and optionally, a country code). For example: ConsentForm_en.pdf for the English version. See How to name consent form files below.
4. Edit the PDF using Adobe Acrobat Professional to include the necessary fields and buttons:
    1. Add a text field wherever you want information to be supplied automatically by Onyx (for example, participant and interviewer information--see the table below).
        1. Set each field to read-only.
        2. Rename each field so it can be automatically completed by Onyx (see the table below).
        3. Apply formatting to all fields that require it (using Acrobat field formatting; for example: date formatting).
    2. Add signature fields where signatures are required. If the signature is required, append "_mandatoryField" to the end of the signature field name.
    3. If the interviewer must enter more information manually, add as many fields as necessary (for example: radio buttons and text fields). If the field is required, append "_mandatoryField" to the end of a field name.
    4. Add one or more form submit buttons.
        1. Add a button to accept the consent. Set the button name to "AcceptURL".
        2. Add a button to refuse the consent. Set the button name to "RefuseURL".
        3. Set each button so they are invisible if the form is printed.
    5. Select ADVANCED -> ENABLE USAGE RIGHTS IN ACROBAT READER in the top menu.
    6. Save the PDF file.

How to name consent form files

The filenames of PDF consent forms must conform to this configurable naming scheme:

`basename_language_country`.pdf

where,

- `basename` is a configurable prefix for the filename (ConsentForm by default). It can be configured in the `WEB-INF/config/marble/module-config.properties` file.

- `language` is the two-letter ISO Language Code (ISO-639-1) code for the form's language (example: `en` for English, `fr` for French). See language codes.
- `country` (optional) is a two-letter ISO Country Code (ISO-3166). See country codes.

Examples of valid consent form names:

- `ConsentForm_en.pdf`

- `ConsentForm_fr.pdf`

- `ConsentForm_fr_CA.pdf`

> When multiple consent stages are defined, you can configure the PDF form basename for each stage using the following format: `org.obiba.onyx.marble.consent.<STAGE NAME>.basename`. Where <STAGE NAME> is the name of the consent stage as defined in the `stages.xml` file.

Where to put consent form files

By default, PDF consent forms should be located in the `WEB-INF/config/marble/resources` directory. Based on the consent form files in the `resources` directory, Onyx "discovers" the available languages and displays them at runtime. No further configuration is required.

You can change the `resources` directory by modifying this line in the `WEB-INF/config/marble/module-config.properties` file:

WEB-INF/config/marble/module-config.properties

```
org.obiba.onyx.marble.resources.path=${org.obiba.onyx.marble.path}/resources
```

How to name fields in PDF forms

At runtime, Onyx can complete certain fields in your consent forms automatically. The mapping between Onyx variables and fields in the PDF form is defined in `WEB-INF/config/marble/module-config.properties`. For a list of the Onyx variables that you can use in your consent form, see To view a list of Onyx variables.

Here is the variable-to-field mapping for the consent form in the onyx-demo application:

WEB-INF/config/jade/module-config.properties

```
[...]
org.obiba.onyx.marble.consent.variableToField=CONTACT_ME=Radio1_mandatoryF
ield,
    SEND_RESULTS=Radio2_mandatoryField

[...]
```

> When multiple consent stages are defined, you can configure different variable-to-field mappings for each consent stage with the following format: `org.obiba.onyx.marble.consent.<STAGE NAME>.variableToField`. Where `<STAGE NAME>` is the name of the consent stage as defined in the `stages.xml` file.

The table below indicates information that Onyx can supply and how you should name the fields in your consent form, if you want Onyx to complete the fields.

How to name the field in consent forms	Notes
Participant.firstName	You can use the Participant attributes listed in this table by simply naming the fields as indicated in the first column. You can include fields that display additional Participant attributes. The attributes must already have been defined for Participant (see). The syntax for naming these fields is: **Participant.attributes[ATTRIBUTE NAME]** For example: Participant.attributes[HomeNumber]
Participant.lastName	
Participant.gender	
Participant.birthDate	
Participant.enrollmentId	
Participant.barcode	This attribute is labelled "participant ID" in the Onyx user interface.
User.firstName	These are the attributes of the User who is currently logged in. In other words, the interviewer who typically signs the consent form as a witness to the participant's signature.
User.lastName	
User.login	Interviewer's username.
User.email	

Customizing Consent Stage Messages

You can modify various texts displayed onscreen during the consent stage by editing the `messages_language.properties` files located in the `WEB-INF/config/marble` directory. The directory contains a `messages` file for each user interface language supported by Onyx. By default, there are `messages files` for English and French.

For example, if you want to change the English name for the stage to be displayed onscreen, you would edit `Consent.description` line in this

file:

WEB-INF/config/marble/messages_en.properties

```
# Stage descriptions
Consent.description=Participant Consent
PrintConsentForm=Print Consent
ConsentFormLabel=Signed Consent
```

Disabling the Consent Stage

If your study does not require that participants sign a consent form, you can disable the consent stage as follows:

1. Remove the `<stage>` element from the `marble/stages.xml` file. The element to remove looks like this:

WEB-INF/config/marble/stages.xml

```
<stage>
    <name>Consent</name>
    <module>marble</module>
</stage>
```

---

2. Remove all the `<variableCondition>` elements related to the consent stage from the `stages.xml` files in the jade and ruby directories. The elements to remove look like this:

WEB-INF/config/jade/stages.xml   and   WEB-INF/config/ruby/stages.xml

```
<variableCondition>
        <stageName>Consent</stageName>
        <variablePath>Consent.accepted</variablePath>
</variableCondition>
```

---

3. Remove all the `<stageDependencyCondition>` elements from the `stages.xml` file in the quartz directory. The elements to remove look like this:

WEB-INF/config/quartz/stages.xml

```
<stageDependencyCondition class="variableCondition">
      <stageName>Consent</stageName>
      <variablePath>Consent.accepted</variablePath>
    </stageDependencyCondition>
```

---

4. Remove the word `Consent` from the `staticStageOrder` property line in the onyx-config.properties file:

WEB-INF/config/onyx-config.properties

```
# A comma separated list of stage names that define their ordering within
an Interview.
org.obiba.onyx.engine.staticStageOrder=Consent,HealthQuestionnaireSelfAdmi
nistered, ...
```

Building Custom Questionnaires

Overview

Your organization probably requires one or more questionnaires for its custom version of Onyx. One of the questionnaires may be a contraindication questionnaire which participants would answer first in order to establish if there are any reasons the participant should not do one of the other stages. All questionnaires (including contraindication questionnaires) are built in the same way. This section explains how to build a questionnaire.

- The Process of Building a Questionnaire
- Question Types
- Questionnaire Terminology
- Planning Your Questionnaire
- Localizing Your Questionnaire
- Styling Your Questionnaire
- Checking Your Questionnaire
- Including Your Questionnaire in Onyx
    - Modifying the Onyx configuration
        - To edit stages.xml
        - To localize the name of your questionnaire

The Process of Building a Questionnaire

To build questionnaires:

1. start your Onyx web application and login as a user having `Questionnaire Editor` rights,
2. go to Administration > Questionnaires,
3. from there you can create, edit, upload questionnaires using the questionnaire editor (see questionnaire section in Onyx User Guide),
4. then configure questionnaire stages,
5. test the questionnaires by interviewing a dummy participant,
6. loop the test-and-edit process until the questionnaires are finally done.

Question Types

Onyx questionnaires can include questions in many formats. You can view screenshots of the different question formats on the OBiBa website.

Questionnaire Terminology

A **category** is one of the possible answers to a question that are proposed to the participant.

An **answer** is the category (or categories) selected by the participant.

An **open-answer question** is a question that the participant responds to by supplying their own answer rather than by selecting a category. The participant's answer is entered in a field.

Planning Your Questionnaire

Before you get deep into programming your questionnaire, we recommend that you and your organization design the required questionnaires as part of your customization plan.

As a minimum, for each question, you must decide the question format, the question text, and the text for the categories. We recommend to define also category names and the variable names that will be associated to questions and open answers.

The questions should be grouped into pages (questions that will appear on the same screen). Questions can also be grouped into sections (by topic).

You can add one or more conditions to a question to control whether the question will be displayed or skipped. The conditions can be:

- the answer to a previous question
- a measurement from an instrument
- a participant attribute
- any combination of the above

You can apply the same condition to several questions in a page or a section. Initially, you could display one question, and only display the other questions if the participant selects a particular answer for the first question. For example, if the participant answers "No" to "Are you pregnant?", then you would not display any of the questions on which this condition had been set: `setCondition("PREGNANT", Y)`.

You must decide which type of screen the questionnaire will be displayed on:

- standard screen
- touchscreen (suitable for self-administered questionnaires)

Localizing Your Questionnaire

Depending on how many locales (languages) you specify in your questionnaire builder program, one or more properties files will have been generated when you run the program. There should be a properties file per locale.

You can add and edit questionnaire localization files using the questionnaire editor. But you can also do it manually by editing the localization text in the files.

For example, if you created your questionnaire in English and French, two properties files should have been generated: `language_en.properties` and `language_fr.properties`.

Each properties file will contain a key for each questionnaire element in your questionnaire. You must ensure that string values are supplied for all keys in all of the properties file.

Styling Your Questionnaire

If you need to overwrite default Onyx style, edit `css/questionnaire-default.css` in the webapp directory.

```
/*
 * Customized default questionnaire elements stylesheet.
 * Use css classes based on questionnaire elements
 (<questionnaire>-<type>-<name>).
 * Use .obiba-lang-* class to customize per questionnaire administration
 language.
 */
@import url("questionnaire-default-obiba.css");


// add your custom styles here
```

Checking Your Questionnaire

You will probably want to write your questionnaire bit by bit, and check that your questionnaire is being structured as you intend. You can check your questionnaire as you go along by generating an HTML version of the questionnaire using the `HTML Preview` feature form the questionnaires list. This saves you the trouble of installing the questionnaire, running Onyx, and answering questions in order to advance through the questionnaire.

Including Your Questionnaire in Onyx

When your questionnaire is complete, the questionnaire files are in the appropriate place in the onyx directory structure: `WEB-INF/config/quartz/resources/questionnaires`

For example, for a questionnaire named "HealthQuestionnaire", version "1.0", localized in English and French, the questionnaire files would go here:

- WEB-INF/config/quartz/resources/questionnaires/HealthQuestionnaire/1.0/questionnaire.xml
- WEB-INF/config/quartz/resources/questionnaires/HealthQuestionnaire/1.0/language_en.properties
- WEB-INF/config/quartz/resources/questionnaires/HealthQuestionnaire/1.0/language_fr.properties

For a touchscreen questionnaire that includes image categories, you would put the images in this directory:

- WEB-INF/config/quartz/resources/questionnaires/HealthQuestionnaire/1.0/images

If several versions of the same questionnaire exist, the one with the highest version number is active. The version numbers must respect this pattern of digits separated by dots: "1.0", "1.0.1", "2.1.0", and so on.

Modifying the Onyx configuration

You must add your custom questionnaire stage to Onyx and remove any of the questionnaire stages that came with the onyx-demo that you do

not want.

To edit stages.xml

If you want to define any stage dependencies, see Defining Stage Dependencies.

You should also remove any stages that you do not want in your custom version of Onyx

---

WEB-INF/config/quartz/stages.xml

```
<stages>
   <stage>
      <name>HealthQuestionnaire</name>
      <module>quartz</module>
   </stage>
</stages>
```

To localize the name of your questionnaire

You must also edit each of the properties files in the `WEB-INF/config/quartz` directory to localize the names of the questionnaire in each interface language.

For example:

In the English properties file (`WEB-INF/config/quartz/messages_en.properties`), add a line like this:

HealthQuestionnaire.description=Custom Health Questionnaire

In the French properties file (`WEB-INF/config/quartz/messages_fr.properties`), add a line like this:

HealthQuestionnaire.description=Questionnaire personnalisé sur la santé

Creating and Editing Questionnaires

Overview

This section explains how to build a questionnaire using the questionnaire editor graphical user interface.

Creating and Editing Questionnaires

This chapter covers creating and editing questionnaires. Typically, this work is done by researchers. To create and edit questionnaires, you must have been assigned the user role of Questionnaire Editor. See User Roles.

This chapter covers:

- Key concepts related to creating and editing questionnaires
- Procedures for creating and editing questionnaires

Key Concepts

This section contains concepts related to creating and editing questionnaires.

Questionnaire

A questionnaire is a set of questions and the possible responses associated with each question. A questionnaire can be used as a stage in an interview. The questions can be of different types. A questionnaire is stored in an XML file that defines the structure of the questionnaire and includes information about how the questions will be displayed onscreen. Questionnaires can be downloaded. The XML file that defines the questionnaire and the properties files that contain the labels (one per display language) are output to a zip file. They can also be uploaded: a zip file containing the XML file and properties files for a questionnaire is decompressed, and the questionnaire is opened in Onyx.

Section

A section is used to group questions by topic. A questionnaire must have at least one section. See Adding a Section.

Page

A page is used to group questions on the same screen. A section must have at least one page. See Adding a Page.

Question Types

A question is used to ask the participant for some information or to present some information. Several types of question are available. See the table for descriptions of the question types.

Question Types

Instruction	A general instruction or explanatory text about a section or a page. See Adding an Instruction.
Single open answer	A question that the participant responds to by supplying an answer rather than by selecting a category. The participant's answer is entered in a field. The answer has a data type ( ). Optionally, a list of possible values can be displayed for the participant to choose from. See Adding Single Open Answer Questions.
List (radio buttons)	An exclusive choice question whose categories are displayed as radio buttons.The participant can only choose one category. See Adding a List Question.
List (drop-down)	An exclusive choice question whose categories are displayed in a drop-down list. The participant can only choose one category.See Adding a List Question.
List (checkbox)	A multiple choice question whose categories are displayed with checkboxes so the participant can select all categories that apply. The participant can select several categories. See Adding a List Question.

Array (radio buttons)	An array question includes several child questions. The child questions automatically share the categories of the parent question. Using radio buttons forces the participant to choose only one category for each child question. The array is displayed as a table in which the rows are questions and the columns are categories. See Adding an Array Question.
Array (checkbox)	An array question includes several child questions. The child questions automatically share the categories of the parent question. Using checkboxes allows the participant to select all the categories that apply. The array is displayed as a table in which the rows are questions and the columns are categories. See Adding an Array Question
Single audio recording	A question where the supplied answer of the participant can be recorded. The interviewer can manually start/stop/pause the recording. A sampling rate and a maximum recording time can be set. See Adding Single Audio Recording Questions.

Category

A category is one of the possible answers to a question that are proposed to the participant. Unlike an open answer, a category does not have a data type. See Defining Categories.

Open Answer

An open-answer question requires the participant to supply an answer rather than select a category. An open answer question has a data type (date, decimal, integer, or text). The participant's answer is usually entered in a field. Optionally, a list of possible answers can be displayed for the participant to choose from.

An open answer field can also be added to a category. This allows further information to be stored with the category. For example, if a category is "Other", an open answer would allow the participant to be more precise.

See Adding Single Open Answer Questions, Adding an Open Answer to a Category, Adding a Single Auto-Complete Opan Answer, Adding an Auto-Complete Opan Answer to a Category and Adding an Audio Recording to a Category.

Condition

You can add one or more conditions to a question to control whether the question will be displayed or skipped. See Adding Conditions to a Question.

Variable

Variables store discrete bits of data relevant for a study. A variable is automatically created for each question added to a questionnaire. You can create additional variables explicitly. See Adding a Variable. Onyx has predefined variables for participant identification information. Variables can be included in conditions and in validators for open answers.

When Onyx automatically creates a variable for a question, it generates a fully qualified name for the variable, meaning the name includes prefixes for the section and page, as well as the question name. You can change these generated names to make them shorter or more relevant for analysis.

Script

A script is a list of one or more commands that are evaluated as you build your questionnaire or when the questionnaire is being completed. A script is always associated with a variable. You write scripts when you create a custom variable, add conditions to a question, and define validators for open answers.

Scripts are written using Javascript syntax and the Magma API (Application Programming Interface). For more information see the Magma API documentation.

Attribute

An attribute is a metadata used to describe a questionnaire element. Questions, categories and open answers can be annotated by some attributes. An attribute is defined by a namespace (optional), a name (required) and a value (required). This value can be optionally localized (i.e. can have a different value for different languages). See Adding Attributes to a Questionnaire Element.

Annotating questionnaire elements with attributes will affect the associated data dictionary (i.e. the questionnaire variables that will be exported from Onyx). See the documentation on Variables for more details.

From a questionnaire question, several variables can be extracted. For instance in the case of a question with multiple choices, one variable per category is defined (that tells whether or not the category was chosen). If the question associated to the category is annotated with attributes, each of the category variables will inherit from the question attributes. This applies also to open answers which are, in the Onyx questionnaire

model, always associated to a category, which is itself in a question. Therefore the variable associated to the open answer will inherit from the attributes of the parent question and category. In case of same attributes (meaning same namespace and name and locale) are defined in the parent question and category, the most specific one is applied to the variable (open answer attribute will override the one from the parent category etc.).
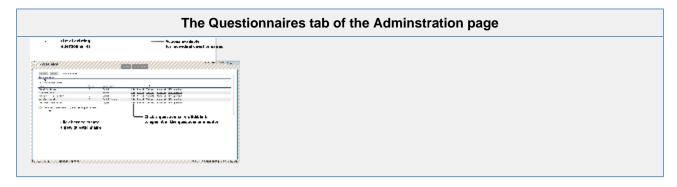
Procedures

This section contains procedures for creating questionnaires and adding all the components necessary to make them fully functional, as well as the procedure for putting them online in Onyx.

To carry out the procedures in this section, you must have been assigned the user role of Questionnaire Editor. See User Roles.

Accessing the Questionnaires Page

You create and edit questionnaires starting from the `Questionnaires` tab of the Onyx `Administration` page. See this figure. You access the `Questionnaires` tab as explained here.

The Questionnaires tab of the Adminstration page

Procedure

1. Log in to the Onyx. See How to log in.
2. Select the `Administration` link in the upper right corner of any Onyx page. By default, the administration page opens on the `User` tab. If you do not see the `Administration` link in the upper right corner of the page, ask the system administrator to assign you the user role of Questionnaire Editor.
3. Select the `Questionnaires` tab. The list of existing questionnaires is displayed.

Creating a New Questionnaire

Before creating a questionnaire, you should probably know the display languages for your questionnaire (for example, English and French) and be ready to provide text in each of those languages for labels that will be displayed throughout the questionnaire. If necessary, more display languages can be added later.

Procedure

This procedure assumes you are in the `Questionnaires` tab of the Onyx `Administration` page. You can access it as explained in Accessing the Questionnaires Page.
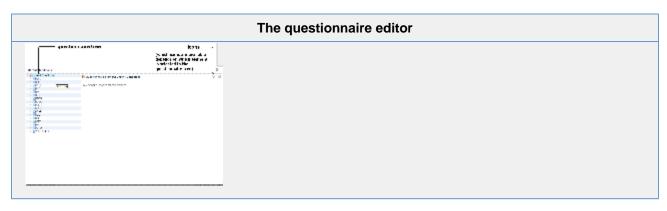
1. Click the `Create new questionnaire` button. The `New questionnaire` page pops up.
2. Enter a `Name` for your questionnaire.
3. Enter a `Version` for your questionnaire.
4. Select a `Display mode`:
   - `Standard` - the type of screen used with desktop computers or the screen of a laptop computer. You select onscreen objects with a mouse or a touchpad.
   - `Touch-screen` - allows you to select buttons and other onscreen objects by touching them with a finger.
5. Select one or more `Languages` for the questionnaire interface. A tab is displayed for each of the languages you selected.
6. Complete all of the fields on each language tab with text in the appropriate language:
   - `label`: The title of the questionnaire that will be displayed. Required.
   - `description`: A brief description of the questionnaire (in the selected language). Optional.
   - In the rest of the fields on this page (`labelNext`, `labelPrevious`, and so on), enter labels (in the selected display language) for the navigation buttons that will appear throughout the questionnaire.
     If these fields are not completed, the default English text will be used.
7. Click the `Save` button. The questionnaire editor pops up with the name of your questionnaire on the left side and help text on the right side.

Before adding questions to your new questionnaire, you must add a section and a page to it. See Adding a Section and Adding a Page.

⚠ *Editing a Questionnaire*

You edit questionnaires in a page called the questionnaire editor. See this figure. You open an existing questionnaire in the questionnaire editor as explained here.

The questionnaire editor

1. If the list of questionnaires is not displayed, access it as explained in in Accessing the Questionnaires Page.
2. Click the `Edit` link for the questionnaire you wish to edit. The questionnaire editor pops up. A tree representing the structure of the questionnaire is displayed in the left panel. Initially, a help page is displayed in the right panel.
3. To start editing the questionnaire, click the name of the the questionnaire (at the top of the tree). The help page is removed, and some icons are displayed near the top right corner of the editor page. The icons displayed depend on which element is selected in the questionnaire tree.

⚠ *Adding a Section*

The questions in a questionnaire can be grouped into sections by topic. Even if you do not want to or need to group questions by topic, you must add at least one section to your questionnaire as explained in this procedure.

If you want to use subsections, you can do so by adding sections to sections.

Before you can add questions to a section, you must add a page to it as explained in Adding a Page to a Section.

Procedure

This procedure assumes the questionnaire is displayed in the questionnaire editor. To open a questionnaire in the questionnaire editor, see Editing a questionnaire.

1. To add a top-level section to your questionnaire, select the name of your questionnaire in the tree on the left.
2. To add a subsection to an existing section, open the questionnaire tree and select the section to which you want to add a subsection.
3. Click the `Add Section` icon in the upper right corner of the dialog. Fields for entering section properties are displayed in the right panel.
4. Enter a name for the section. This name is used internally and will not be displayed.
5. Enter a label for the section in each of the language tabs. This label will be displayed so the text must be in the appropriate display language.
6. Click the `Save` button. The new section is added to the questionnaire tree.

⚠ *Adding a Page*

Questions can be grouped into pages to ensure they will appear on the same screen. You can also use pages to ensure that only one question is displayed per screen. Even if you do not care how many questions are displayed per screen, you must add at least one page to each section of your questionnaire as explained in this procedure. After adding a page, you can select it and add questions to it.

Procedure

This procedure assumes the questionnaire is displayed in the questionnaire editor. To open a questionnaire in the questionnaire editor, see Editing a questionnaire.

1. Open the tree for your questionnaire until you find the section to which you want to add a page.
2. Click the `Add Page` icon in the upper right corner of the dialog. Fields for entering page properties are displayed in the right panel.
3. Enter a name for the page. This name is used internally and will not be displayed.
4. Enter a label for the page in each of the language tabs. This label will be displayed so the text must be in the in the appropriate display

language.
5. Click the **Save** button (at the bottom of the page---you may have to scroll down).

⚠ *Adding an Instruction*

You can add directions to users before and between questions with a special type of question called an instruction. It is simply text.

Procedure

This procedure assumes the questionnaire is displayed in the questionnaire editor. To open a questionnaire in the questionnaire editor, see Editing a questionnaire.

1. Open the tree for your questionnaire until you find the section and page in which you want to add the instruction.
2. Click the **Add Question** icon in the upper right corner of the dialog.
3. Enter a name for the instruction in the **Name** field. This name is used internally and will not be displayed.
4. Select **Instruction** in the **Type** field.
5. For each language, enter the actual instruction in the **label** field.
6. Click the **Save** button.

⚠ *Adding a Single Open Answer Question*

A single open answer question displays the question label and a single field in which to enter the participant's response. An open answer has a data type (date, decimal, integer, or text).

As an option, you can define a unit (e.g.'mm' for millimeters or 's' for seconds), a field size (width of the textfield) and a number of lines (only applicable for text open answers).

Also, you can define a drop-down list of possible values that the participant can choose from. The answer selected is stored as a value with a data type. This is different from a List question that displays a drop-down list, in which case the participant's selection is stored as a category and does not have a data type.

Procedure

This procedure assumes the questionnaire is displayed in the questionnaire editor. To open a questionnaire in the questionnaire editor, see Editing a questionnaire.

1. Open the tree for your questionnaire until you find the section and page in which you want to add the question.
2. Click the **Add Question** icon in the upper right corner of the dialog. Fields for entering question properties are displayed in the right panel.
3. Enter a name for the question in the **Name** field. This name is used internally and will not be displayed.
4. If you wish, enter a custom name for the variable associated with the question in the **Variable** field. This variable name is used internally and will not be displayed.
5. Select **Single open answer** in the **Type** field.
6. In each of the language tabs, enter the text that will appear onscreen in the appropriate language. For more information, see Entering Language Properties for a Question.
7. Click the **Save** button.

⚠ *Adding a List Question*

List questions are used for questions that have a set of categories as possible answers. Three types of list question are available:

- **Radio button list**. Exclusive choice question (participant can select only one category), suitable for a small number of categories. The categories can be displayed in different layouts (default, single column or grid).
- **Drop-down list**. Exclusive choice question (participant can select only one category), suitable for a large number of categories.
- **Checkbox list**. Multiple choice question (participant can select several categories). The categories can be displayed in different layouts (default, single column or grid), whichever is most suitable for the number of categories.

You can add categories to a list question one at a time or add several categories at once. You can also add existing categories (categories defined for another question) to a list question.

You can obtain more detail from a participant who selected a particular category by adding an open answer to the category. See Adding an Open Answer to a Category.

After defining a list question, you can change the question type to another type of list question if necessary.

Procedure for adding a list question

This procedure assumes the questionnaire is displayed in the questionnaire editor. To open a questionnaire in the questionnaire editor, see Editing a questionnaire.

1. Open the tree for your questionnaire until you find the section and page in which you want to add the question.
2. Click the `Add Question` icon in the upper right corner of the editor. Fields for entering question properties are displayed in the right panel.
3. Enter a name for the question in the `Name` field. This name is used internally and will not be displayed.
4. If you wish, enter a custom name for the variable associated with the question in the `Variable` field. This variable name is used internally and will not be displayed.
5. Select the question type that best suits your needs from the list in the `Type` field:
   - `List (radio)` - for an exclusive choice question with a few categories
   - `List (drop-down)` - for an exclusive choice question with more than a few categories
   - `List (checkbox)` - for a multiple choice question
6. In each of the language tabs, enter the text (in the appropriate language) that will appear onscreen. For more information, see Entering Language Properties for a Question.
7. Click the `Categories` tab.
8. If you do want not to use the default layout (a grid with five rows) for categories, select another option in the `Layout` list. Here are some pointers:
   - For a checkbox list, you might want to select `Single column` since the default layout will use several columns if there are enough categories.
   - `Grid` layout is useful if you want to control how radio buttons or checkboxes are displayed in columns.
9. Add categories to the question. For detailed directions, see these procedures:
   - Adding a Single Category to a List Question
   - Adding Multiple Categories to a List Question
   - Adding Exising Categories to a List Question
10. Complete the definition of the categories (adding labels and so on). See Defining the Properties of a Category.
11. Click the `Question` tab and enter the order in which you want the categories to be displayed in the `categoryOrder` field at the bottom of the editor. Enter the category names as a comma-separated list. For example: `category3,category2,category1`.
12. Click the `Save` button.

Procedure for adding a single category to a list question

This procedure assumes that your questionnaire is open in the questionnaire editor, that you are editing a particular list question, and that you are in the `Categories` tab for that question.

1. Select the `Simple add` tab.
2. Enter a name for the category in `New category` field.
3. Click the + button (to the right of the `New category` field). The category is added to the list of categories.
4. Repeat the last two steps if you want to add one more question. Or carry out the next procedure to add several questions at once.
5. Click the `Save` button.

If you wish, complete the definition of the categories (labels and so on) immediately. See Defining the Properties of a Category.

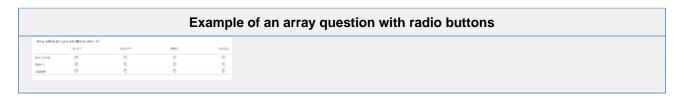**Procedure for adding multiple categories to a list question**

This procedure assumes that your questionnaire is open in the questionnaire editor, that you are editing a particular list question, and that you are in the `Categories` tab for that question.

1. Click the `Multiple add` tab.
2. Enter a comma-separated list of names for the new categories in the `New categories` field. For example: category1,category2,category3.
3. Click the + button (to the right of the `New categories` field). The categories are added to the list of categories.
4. Click the `Save` button.

If you wish, complete the definition of the categories (labels and so on) immediately. See Defining the Properties of a Category.

Procedure for adding existing categories to a list question

This procedure assumes that your questionnaire is open in the questionnaire editor, that you are editing a particular list question, and that you are in the `Categories` tab for that question.

1. Click the `Add existing categories` tab.
2. Locate and select an existing category as follows:
   - Enter part or all of the name of an existing category in the `Category name` field. As you type, one or more existing categories will be displayed in a popup list.
   - When the category you are looking for appears in the popup list, move the cursor over it, and wait for the category to be highlighted in yellow.
   - Click in the name of the desired category. The name of the category is inserted in the `Category name` field.
3. Click the + button (to the right of the `Category name` field). The category is added to the list of categories.
4. Repeat the last two steps for each existing category that you want to add.
5. Click the `Save` button.

Adding an Array Question

Array questions are used for several questions that take the same set of categories. An array question is displayed as a table: the questions are displayed as rows, and the categories are displayed as columns. Use a radio button array list if only one category should be selected for each question. Use a checkbox array list if several categories can be selected for each question.

Example of an array question with radio buttons

Procedure for adding an array question

This procedure assumes the questionnaire is displayed in the questionnaire editor. To open a questionnaire in the questionnaire editor, see Editing a questionnaire.

1. Open the tree for your questionnaire until you find the section and page in which you want to add the question.
2. Click the `Add Question` icon in the upper right corner of the dialog. Fields for entering question properties are displayed in the right panel.
3. Enter a name for the question in the `Name` field. This name is used internally and will not be displayed.
4. If you wish, enter a custom name for the variable associated with the question in the `Variable` field. This variable name is used internally and will not be displayed.
5. Select the question type that best suits your needs from the list in the `Type` field:
    * `Array (radio)` - if participants can only select one category for each question
    * `Array (checkbox)` - if participants can select several categories for each question
6. In each of the language tabs, enter the text (in the appropriate language) that will appear onscreen. For more information, see Entering Language Properties for a Question.
7. At this point, you can add questions (rows) and categories (columns) to the array question. For detailed directions, see these procedures:
    * Adding a Single Question (Row) to an Array Question
    * Adding Multiple Questions to an Array Question
    * Adding a Single Category to an Array Question
    * Adding Multiple Categories to an Array Question
    * Adding Existing Categories to an Array Question
8. Click the `Question` tab and enter the order in which you want the categories to be displayed in the `categoryOrder` field at the bottom of the dialog. Enter the category names as a comma-separated list. For example: `category1,category2,category3`.
9. Click the `Save` button.

Procedure for adding a single question (row) to an array question

This procedure assumes that you have opened your questionnaire in the questionnaire editor and that you are editing a particular array question.

1. Click the `Rows (questions)` tab.
2. Click the `Simple add` tab.
3. Enter a name for the question in the `New question` field.
4. Click the `+` button (to the right of the `New question` field). The question is added to the list of questions.
5. Repeat the last two steps if you want to add one more question. Or carry out the next procedure to add several questions at once.
6. If you wish, you can complete the definition of the categories immediately. See Defining the Properties of a Category.

Procedure for adding multiple questions (rows) to an array question

This procedure assumes that you have opened your questionnaire in the questionnaire editor and that you are editing a particular array question.

1. Click the `Rows (questions)` tab.
2. Click the `Multiple add` tab.
3. Enter comma-separated list of names for the questions in the `New questions` field. For example, `question1`, `question2`,`question 3`.
4. Click the `+` button (to the right of the `New questions` field). The questions are added to the list of questions.
5. If you wish, complete the definition of the categories immediately. See Defining the Properties of a Category.

Procedure for adding a single category (column) to an array question

This procedure assumes that you have opened your questionnaire in the questionnaire editor and that you are editing a particular array question.

1. Click the `Columns (categories)` tab.

2. Click the `Simple add` tab.
3. Enter a name for the category in the `New category` field.
4. Click the `+` button (to the right of the `New category` field). The category is added to the list of categories.
5. Repeat the last two steps if you want to add one more category. Or carry out the next procedure to add several categories at once.
6. If you wish, you can complete the definition of the categories immediately. See Defining the Properties of a Category.

Procedure for adding multiple categories (columns) to an array question

This procedure assumes that you have opened your questionnaire in the questionnaire editor and that you are editing a particular array question.

1. Click the `Columns (categories)` tab.
2. Click the `Multiple add` tab.
3. Enter a comma-separated list of names for the categories in the `New categories` field.
4. Click the `+` button (to the right of the `New categories` field). The categories are added to the list of categories.
5. If you wish, you can complete the definition of the categories immediately. See Defining the Properties of a Category.

Procedure for adding existing categories to an array question

This procedure assumes that you have opened your questionnaire in the questionnaire editor and that you are editing a particular list question.

1. Click the `Columns (categories)` tab.
2. Click the `Add existing categories` tab.
3. Locate and select the category as follows:
   - Enter part or all of the name of an existing category in the `Category name` field. As you type, one or more existing categories will be displayed in a popup list.
   - When the category you are looking for appears in the popup list, move the cursor over it, and wait for the category to be highlighted in yellow.
   - Click in the name of desired category. The name of the category is inserted in the `Category name` field.
4. Click the `+` button (to the right of the `Category name` field). The category is added to the list of categories.

Repeat the above steps for each existing category that you want to add.

Adding a Single Auto-Complete Open Answer

An auto-complete open answer allows user to enter an open value with the help (or the constraint) of auto-completed values. The list of choices is configurable and can change over time. The list of choices can be extracted from a static list or from the values of a variable.
Auto-complete works so that when the user writes the first letter of a word, Onyx predicts one or more possible words as choices.
The suggestion list building

- is case insensitive
- try to match the start of all items words, where word separators are one or more occurences of the characters: (space) , ; / . : - _ # | ( ) [ ] { }
- stop when maximum suggestion count is reached

For instance, if the choices are drug names and the input string is **ca**, suggestions will be:

- **CANESTEN** COMBI-PAK CREAM 1
- **CARMEX** FLAVOURED LIP BALM
- APO-DOXY **CAP** 100MG
- DILTIAZEM-CD 180MG **CAPSULES**

The data type of the auto-complete open answer is always **text**.

Procedure

This procedure assumes the questionnaire is displayed in the questionnaire editor. To open a questionnaire in the questionnaire editor, see Editing a questionnaire.

1. Open the tree for your questionnaire until you find the section and page in which you want to add the question.
2. Click the `Add Question` icon in the upper right corner of the dialog. Fields for entering question properties are displayed in the right panel.
3. Enter a name for the question in the `Name` field. This name is used internally and will not be displayed.
4. If you wish, enter a custom name for the variable associated with the question in the `Variable` field. This variable name is used internally and will not be displayed.
5. Select **Single auto-complete** in the **Type** field.
6. In each of the language tabs, enter the text that will appear onscreen in the appropriate language. For more information, see Entering Language Properties for a Question.
7. Select the **Auto-complete** tab and enter a name for the open answer in the **Name** field.
8. If needed, you can enter a unit of measure in the **Unit** field.

9. If you want to specify the width of the field that will be displayed, you can enter a value in the **Field size** field.
10. You can limit the size of the suggestion list by entering a value in the **Max number of suggestions** field.
11. Select the source of suggestions
    1. For auto-complete based on a list of items
        1. Select radio **List of items**
        2. Enter a single item using **Simple add** tab or use **Multiple add** tab to enter several items (comma-separated list of names). These items can be associated with a label in a given language, in which case the item label is displayed and the item value is stored. If no label is defined, the item value is displayed.
        3. Click on the **Edit** link to edit the item name and label.
    2. For auto-complete based on variable values
        1. Select radio **Variable values**
        2. Select the **Datasource** and **Table** that contain the variable you want to use.
        3. Select the **Variable** for a given language
12. In each of the language tabs, enter the text that will appear onscreen in the appropriate language. For more information, see Entering Language Properties for a Question.
13. Auto-complete validation:
    1. Check **Required** checkbox if the user must enter a value for this open answer.
    2. Select **Allow suggested text only** radio to restrict entered value to suggested items only.
    3. Select **Allow open text** to allow user to enter a value that was not suggested. To validate the open answer by comparing it to a pattern, enter a regular expression in the **Pattern** field.
14. Click the **Save** button.

Adding a Single Audio Recording Question

A single audio recording question displays the question label and a single field in which the interviewer has the control to start/pause/stop recording. An open audio recording has a data type 'binary'.

Procedure

This procedure assumes the questionnaire is displayed in the questionnaire editor. To open a questionnaire in the questionnaire editor, see Editing a questionnaire.

1. Open the tree for your questionnaire until you find the section and page in which you want to add the question.
2. Click the **Add Question** icon in the upper right corner of the dialog. Fields for entering question properties are displayed in the right panel.
3. Enter a name for the question in the **Name** field. This name is used internally and will not be displayed.
4. If you wish, enter a custom name for the variable associated with the question in the **Variable** field. This variable name is used internally and will not be displayed.
5. Select **Single audio recording** in the **Type** field.
6. In each of the language tabs, enter the text that will appear onscreen in the appropriate language. For more information, see Entering Language Properties for a Question.
7. Select the **Audio recording** tab and enter a name for the audio recording in the **Name** field. If you wish, you can change the sampling rate and maximum recording time.
8. Click the **Save** button.

Defining Language Properties for a Question

The language properties for a question define the text that will be displayed when the questionnaire is being completed. The label is required. The other properties are optional.

Procedure

This procedure assumes the questionnaire is displayed in the questionnaire editor. To open a questionnaire in the questionnaire editor, see Editing a questionnaire.

1. If the **Language properties** of the questionnaire are not displayed, click the name of the questionnaire in the tree, and then click the **Edit** icon at the right side of the page.
2. In the **Language properties** panel, enter the actual question in the **label** field. This field is required. It is the text that will be displayed in the questionnaire.
3. Optionally, enter instructions to the participant in the **instructions** field. This text will be displayed just below the label.
4. Optionally, enter the media to be displayed to the participant in the **media** field. The specified media could be an image, audio or video stream. Onyx automatically recognizes the following commonly used formats: .png, .jpg, .jpeg, .wav, .oga, .m4a, .mp3, .webm, .ogv, .mp4. For other formats, you must specify the mime-type (e.g. instructions_en.aiff|audio/x-aiff).
5. Optionally, enter additional explanatory text in the **caption** field. This text will be displayed just below the instructions.
6. Optionally, enter help text in the **help** field. This text will be displayed if the user clicks the question mark button beside the label.
7. Optionally, enter additional explanatory text in the **specifications** field. This text will be displayed just below the field for entering the participant's response.
8. Click the **Save** button.

Defining the Properties of a Category

After you add categories to a list or array question, you must enter labels for them and can define a few other properties.

This procedure assumes that categories have already been added to the question and that the list of categories is displayed in the questionnaire editor.

Procedure

1. Click the edit icon for the category (at the right end of the row for the category).
2. For each language tab in the **Language properties** panel, enter text in the appropriate language in the **label** field.
3. If you wish, enter a custom name for the variable associated with the category in the **Variable** field. This variable name is used internally and will not be displayed.
4. Check the **Escape/missing** checkbox if the category represents an answer that means "Don't know" or "Prefer not to answer".
5. Check the **No answer** checkbox if the category represents an answer that means "Not applicable" (for example, a man's answer to "Have you ever been pregnant?").
6. If you wish, you can add an open answer field to the category (for example "Other, please specify"). See Adding an Open Answer to a Category.
7. Click the **Save** button.

Adding an Open Answer to a Category

You may wish to add an open answer to a category. For example, if the category is "Other", you may want to allow the participant to be more specific. The participant's response can be entered in a text field that has a data type (date, decimal, integer, or text).

As an option, you can define a unit (e.g.'mm' for millimeters or 's' for seconds), a field size (width of the textfield) and a number of lines (only applicable for text open answers).

If you want to limit the values that the participant can enter for an open-answer, you can define a list of items that will be displayed in a drop-down list (for example, the days of the week). You can add the items one at a time in the "Single add" tab, or you can add several items at once in the "Multiple add" tab. To validate the selected option, see Validating Open Answer Drop-Down List.

Procedure

This procedure assumes that the category has already been added to the question and that you are editing the category. If you need help getting to this point, see Defining the Properties of a Category.

1. Click the **Add open answer** button (in the lower left corner of the category editor).
2. Enter a name for the open answer in the **Name** field. This name is used internally and will not be displayed.
3. If you wish, enter a custom name for the variable associated with the open answer in the **Variable** field. This variable name is used internally and will not be displayed.
4. Select the appropriate type in the **Data type** dropdown list. The available types are **Date**, **Decimal**, **Integer**, and **Text**.
5. If the **Data type** you selected is numerical, you can enter a unit of measure in the **Unit** field.
6. If you want to specify the width of the field that will be displayed, you can enter a value in the **Field size** field.
7. If you want to define a drop-down list of items that the participant can choose from, add the list items as follows:
   - To add one item, select the **Single add** tab, enter a name for the item in the **New list item** field and click the + button (to the right of the **New list item** field).
   - To add several items at once, select the **Multiple add** tab, enter a comma-separated list of names for the items in the **New list items** field and click the + button (to the right of the **New list items** field).
   - Enter labels in the appropriate language for each tab in the **Language properties** panel.
8. If desired, you can validate open answers, you must define a validator. See Validating Open Answers.
9. Click the **Save** button.

Adding an Auto-Complete Opan Answer to a Category

You may wish to add an auto-complete answer to a category.

Procedure

This procedure assumes that the category has already been added to the question and that you are editing the category. If you need help getting to this point, see Defining the Properties of a Category.

1. Click the **Add auto-complete answer** button (in the lower left corner of the category editor).
2. Enter a name for the open answer in the **Name** field. This name is used internally and will not be displayed.
3. If you wish, enter a custom name for the variable associated with the question in the **Variable** field. This variable name is used internally and will not be displayed.
4. If needed, you can enter a unit of measure in the **Unit** field.

5. If you want to specify the width of the field that will be displayed, you can enter a value in the **`Field size`** field.
6. You can limit the size of the suggestion list by entering a value in the **Max number of suggestions** field.
7. Select the source of suggestions
    1. For auto-complete based on a list of items
        1. Select radio **List of items**
        2. Enter a single item using **Simple add** tab or use **Multiple add** tab to enter several items (comma-separated list of names). These items can be associated with a label in a given language, in which case the item label is displayed and the item value is stored. If no label is defined, the item value is displayed.
        3. Click on the **Edit** link to edit the item name and label.
    2. For auto-complete based on variable values
        1. Select radio **Variable values**
        2. Select the **Datasource** and **Table** that contain the variable you want to use.
        3. Select the **Variable** for a given language
8. In each of the language tabs, enter the text that will appear onscreen in the appropriate language. For more information, see Entering Language Properties for a Question.
9. Auto-complete validation:
    1. Check **Required** checkbox if the user must enter a value for this open answer.
    2. Select **Allow suggested text only** radio to restrict entered value to suggested items only.
    3. Select **Allow open text** to allow user to enter a value that was not suggested. To validate the open answer by comparing it to a pattern, enter a regular expression in the **Pattern** field.
10. Click the **Save** button.

Adding an Audio Recording to a Category

You may wish to add an audio recording answer to a category. For example, if the category is "Name animals beginning with the letter 'C'", you may want to record the participant answer.

You can change the sampling rate and maximum recording time by changing the default values in the **Audio configuration** section.

Procedure

This procedure assumes that the category has already been added to the question and that you are editing the category. If you need help getting to this point, see Defining the Properties of a Category.

1. Click the **`Add audio recording`** button (in the lower left corner of the category editor).
2. Enter a name for the open answer in the **`Name`** field. This name is used internally and will not be displayed.
3. If you wish, enter a custom name for the variable associated with the question in the **`Variable`** field. This variable name is used internally and will not be displayed.
4. If you wish, you can change the sampling rate and maximum recording time.
5. Click the **`Save`** button.

Adding Attributes to a Questionnaire Element

You may wish to add attributes on questions, categories or open answers.

Procedure

1. Click on attributes tab.
2. Click on **Add attribute**.
3. Enter minimally a **name** and a **value** for the attribute. You have to define at least one value for a given locale including the "no locale". Namespace is optional.
4. Click **Save** button

Validating Open Answers

If you have defined an open answer, either in a single open-answer question or as an open answer field associated with a category, you can validate the value entered in the open-answer field. You can validate an open answer in one or more of these ways:

- Comparing the value entered to a pattern (written using regular expression syntax).
  Useful for text. For example, the first 3 characters of a postal code follows this pattern a-zA-Z
  da-zA-Z which means: "a letter" (a-zA-Z) then "a number" (
  d) and then "a letter" (a-zA-Z).
- Defining fixed upper and/or lower limits
- Defining a validator which is essentially a script that compares the value entered to a variable or to an answer to a previous question

You can define an open answer validator for either type of open-answer element as explained in this section. In both cases, the **`Open answer`** de

finition is very similar, and the `Validation` panel is displayed near the bottom of the page.

Procedure

1. If you are adding a validator to a single open answer question, you can display the open-answer definition as follows:
   - Select the question in the questionnaire tree.
   - Click the `Edit question` icon.
   - Select the `Open answer` tab.
2. If you are adding a validator to an open answer field associated with a category, you can display the open-answer definition as follows:
   - Select the question in the questionnaire tree.
   - Click the `Edit question` icon.
   - Select the `Categories` tab.
   - Click the `Edit category` icon for the category that has an open answer associated with it.
   - Click the `Edit` icon for the open answer (near the bottom right of the page).
3. Scroll down to the `Validation` panel of the `Open answer` definition.
4. Check the `Required` checkbox. If you check this box, you must define the validation process as explained in the next three steps.
5. To validate the open answer by comparing it to a pattern, enter a regular expression in the `Pattern` field.
6. To set upper and/or lower limits, complete the appropriate fields (depending on the data type of the open answer, different fields are displayed):
   - `Before` and/or `After` (for data type `Date`; for US, the format is yyyy-MM-dd; for France, the format is dd-MM-yyyy)
   - `Minimum` and/or `Maximum` (for data types `Decimal` and `Integer`)
   - `Min. Length` and/or `Max. Length` (for data types `Text`)
7. To compare the open answer to a variable or an answer to a previous question:
   - Click `Add new validator`.
   - Select an `Operator` (=, <>, <, <=, >. >=, IN). Typically, the **IN** operator allows to evaluate if the selected category was previously selected in another question, see Validating Open Answer Drop-Down List.
   - If you select `Answer to a previous question`, you must then select the `Question` and `Category`.
   - If you select `Variable`, you can select an existing variable from the drop-down list or click the **+** button to add a new variable. If you want to add a new variable, see Adding a Variable.
   - If you select `Script`, you can write a custom script that consists of a logical or mathematical expression. The script is written using Javascript syntax and the Magma API. For more informaiton, see the Magma API documentation. The script can include existing variables.
8. Click the `Save` button.

Validating Open Answer Drop-Down List

You can validate that the option selected in an open answer drop-down list must be the same option as a previously selected category. For example, if you have a question "Select all your sources of income" and then asks for "What is your primary source of income", you may want to validate that the "primary" source of income was selected also in the "all sources of income".

Procedure

This procedure assumes that the open answer has already been added to the question and that you are editing the open answer.

1. You must define a drop-down list of items that the participant can choose from, add the list items as follows:
   - To add one item, select the `Single add` tab, enter a name for the item in the `New list item` field and click the **+** button (to the right of the `New list item` field). It is important that the **New list item** name is the same as the category name that you will want to validate.
   - To add several items at once, select the `Multiple add` tab, enter a comma-separated list of names for the items in the `New list items` field and click the **+** button (to the right of the `New list items` field). It is important that the **New list item** name is the same as the category name that you will want to validate.
   - Enter labels in the appropriate language for each tab in the `Language properties` panel.
2. Click `Add new validator`.
3. Select the IN `Operator`.
4. If you want to validate the selected option with a previously selected option, select `Answer to a previous question`, you must then select the `Question` and `Category`.
5. If you want to validate the selected option with a **Variable**, you can select an existing variable from the drop-down list or click the **+** button to add a new variable. The **Variable** must be **Repeatable**. If you want to add a new variable, see Adding a Variable.
6. Click the `Save` button.

Adding Conditions to a Question

You can add one or more conditions to a question to control whether the question will be displayed or skipped.

Conditions can be defined in three ways:

- Using the answer to a previous question. If the answer to the question was a particular category or open answer, the question is displayed.

- Using a variable. You can select an existing variable or define a new one. At runtime, the script associated with the variable will be evaluated and if the result is true, the question is displayed.
- Using a custom script. You can write a script that consists of a logical or mathematical expression. The script is written using Javascript syntax and the Magma API. For more informaiton, see the Magma API documentation. The script can include existing variables.

Procedure

This procedure assumes the questionnaire is displayed in the questionnaire editor. To open a questionnaire in the questionnaire editor, see Editing a questionnaire.
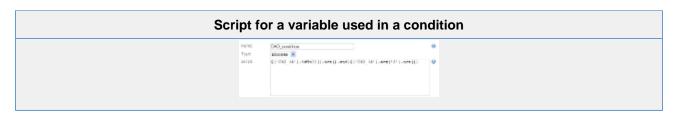
1. Open the questionnaire tree to locate the question to which you want to add a condition.
2. Click on the question.
3. Click the Edit question icon (in the upper right corner of the editor).
4. Select the **Conditions** tab.
5. To base the condition on the answer to a previous question:
    - Select **Answer to a previous question**.
    - Select an question in the **Question** list.
    - Select a category in the **Category** list.
6. To base the condition on a variable:
    - Select **Variable**.
    - To use an existing variable, select one in the drop-down list.
    - To define a new variable and script, select the **+** button beside the drop-down list. For more information, see Adding a New Variable.
    - Select a category in the **Category** list.
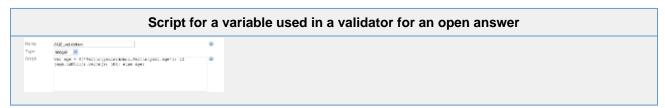7. Click the **Save** button.

 Adding a Variable

Variables are used in conditions (for displaying or skipping questions) and in validators for open answers. Variables are generated automatically when you create questions and categories. Onyx has many predefined variables for patient identification information. If you wish to define custom variables, you can do so as explained in this procedure.

When you add a variable, you write a script that consists of a logical or mathematical expression. The script is written using Javascript syntax and the Magma API, an application programming interface written expressly for biobank data. For more informaiton, see the Magma API documentation. The script can include existing variables (that you can select from a list). At runtime when a user is completing the questionnaire, the script will be evaluated, and the result assigned to the variable. If the variable is part of a condition, Onyx will determine whether or not to display the question. See the example in this figure. If the variable is part of a validator for an open answer, Onyx will accept or reject the value entered by the user. See the example in this figure.

Script for a variable used in a condition

Script for a variable used in a validator for an open answer

Procedure

1. If you are starting from the list of questionnaires:
    - Select the questionnaire in the list of questionnaires.
    - Click the **Edit** link.
    - Click **VARIABLES** in the questionnaire tree.
    - Click the **Add variable** icon.
2. To add or select a variable for use in a validator for an open answer, get to the **Variable** page by following procedure Validating Open Answers.
3. Enter a **Name** for the new variable.
4. Select a **Type** for the new variable. For a condition (used to determine whether a question will be displayed or skipped), select **Boolean**. For an open answer validator, the type should match the type of the open answer.If you entered the variable from the validator for an open answer, you cannot change the data type. The variable is set to the same data type as the open answer.

5. If the variable returns a vector of values, select **Repeatable**.
6. In the `Script` field, enter an expression that will be evaluated at runtime. You can copy and paste existing variables (from the `Existing Variables` panel) in the script, as explained in the next step.
7. If you want to include existing variables:
   - Scroll through the list of `Tables` and select one. Each table corresponds to a questionnaire, except for `Participants` which contains Onyx's predefined variables for participant identification information.
   - Scroll through the list of `Variables` for the selected table and select one. The full name of the variable is displayed.
   - If you want to use the variable in the script, highlight its name in the `Full variable name` field, copy it (Ctrl-C) and paste it (Ctrl-V) into the desired spot in the `Script` field.
8. Click the `Save` button. The variable you just added should appear in the list of variables for the questionnaire the next time you view the questionnaire tree.

Moving Elements in a Questionnaire

After you have defined a certain number of sections, pages, and questions in a questionnaire, you may want to reorganize them. This is easily done by simply selecting the element you want to move in the questionnaire tree and then dragging to a new position in the tree and releasing it. If the move you try to make is not valid, a red X will be displayed when you release the element.

Downloading a Questionnaire

Downloading a questionnaire is the only way of saving a questionnaire for use in another Onyx installation or as a backup for safekeeping.

The output of downloading a questionnaire is a zip file that contains the XML file that defines the questionnaire, and one or more properties files (one for each display language in the questionnaire).

Procedure

1. If the list of questionnaires is not displayed, access it as explained in in Accessing the Questionnaires Page.
2. Click the `Download` link for the questionnaire you wish to download. A dialog pops up.
3. To open the zip file in a viewer, select `Open with`. You can select an application in which to view it.
4. To save the zip file, select `Save`. You can specify a file name and location in which to store it.

Uploading a Questionnaire

Uploading a questionnaire is the way to open a questionnaire that was created in another Onyx installation.

In order to upload a questionnaire, it must have been saved as a zip file that contains the XML file that defines the questionnaire, and one or more properties files (one for each display language in the questionnaire).

Procedure

1. If the list of questionnaires is not displayed, access it as explained in in Accessing the Questionnaires Page.
2. Click the `Upload new questionnaire` button. A dialog pops up.
3. Click the `Browse` button and locate the zip file for the questionnaire that you wish to upload.
4. Select `Save`. If the zip file contains the appropriate for a questionnaire that does not already exist in the current Onyx installation, the questionnaire is added to list of questionnaires.

Configuring Physical Measurement Stages

Overview

Your organization may require one or more physical measurement stages for its custom version of Onyx. Typically, a separate Onyx stage is defined for each type of physical measurement.

The onyx-demo web application includes several physical measurement stages. You can use these stages as models for the physical stages of your custom version of Onyx.

- Configuration Files for a Physical Measurement Stage
- Physical Measurement Stages Copied from onyx-demo
- Defining the Instrument Descriptor File
  - Example of an Instrument Descriptor File
  - Elements of an Instrument Descriptor File
  - Using Data Sources in an Instrument Descriptor File
  - Using Integrity Checks in an Instrument Descriptor File
- Localizing a Physical Measurement Stage

Configuration Files for a Physical Measurement Stage

The configuration files for physical measurement stages are in the `WEB-INF/config/jade/resources/instruments` directory. This directory includes a subdirectory for each physical measurement stage. For example, the subdirectory for the blood pressure stage in onyx-demo is:

```
WEB-INF/config/jade/resources/instruments/BloodPressure
```

The subdirectory for a particular physical measurement stage includes a descriptor file that defines the stage (instrument attributes, input and output parameter attributes, validation, contraindications, and so on) and properties files for localizing the text that appears onscreen. The files are named as follows:

```
instrument-descriptor.xml
messages_en.properties
messages_fr.properties
```

The onyx-demo application includes a couple of physical measurement stages that launch software for particular models of instrument. If an instrument has its own software that will be launched automatically by Onyx, an additional directory will be defined for the instrument in an `instruments` directory at the same level as `WEB-INF`. It is not currently possible to define a custom stage that will launch an instrument's software and capture the data automatically.

Physical Measurement Stages Copied from onyx-demo

When you copied the directory structure of the onyx-demo web application, you got a number of physical measurement stages. There should a subdirectory for the following stages in your `WEB-INF/config/jade/resources/instruments` directory:

```
BloodPressure
Impedance310
Spirometry
StandingHeight
Weight
```

To prepare a new physical measurement stage, you can use these stages as models. If you want to keep any of these stages, you can use them as is or modify them as necessary. A couple of the stages launch proprietary software for particular instrument models. You can use these stages as is in your custom version of Onyx if you wish.

Support has also been implemented for a number of other instruments with proprietary software. Check the complete list of supported instruments, and contact us if you need more information. It is not currently possible to define a custom stage for electronic instruments other than those in the supported list.

Defining the Instrument Descriptor File

The instrument descriptor file defines how a physical measurement stage will be presented in Onyx. To prepare the instrument descriptor file for a custom physical stage, start with a descriptor file for one of the physical measurment stages in onyx-demo and edit the elements as necessary.

Example of an Instrument Descriptor File

An instrument descriptor file defines how the physical measurement stage will proceed in Onyx. The following example of an instrument descriptor file is for a sitting height measurement.

instrument-descriptor.xml

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<instrumentType>
```

```xml
<name>SittingHeight</name>

<properties>
  <property name="org.obiba.onyx.some.property" value="newValue"/>
</properties>

<contraindications>
  <contraIndication>
    <code>UnableToSitUnassistedCast</code>
    <type>OBSERVED</type>
  </contraIndication>

  <contraIndication>
    <code>Other_CI</code>
    <requiresDescription>true</requiresDescription>
    <type>OBSERVED</type>
  </contraIndication>
</contraindications>

<instrumentParameters>
  <interpretative>
    <code>IV_KYPHOSIS</code>
    <dataType>TEXT</dataType>
    <captureMethod>MANUAL</captureMethod>
    <type>OBSERVED</type>
  </interpretative>

  <input>
    <code>INPUT_STOOL_HEIGHT</code>
    <dataType>DECIMAL</dataType>
    <measurementUnit>cm</measurementUnit>
    <captureMethod>AUTOMATIC</captureMethod>
    <dataSource class="scriptDataSource">
      <valueType>decimal</valueType>
      <script>39.5</script>
    </dataSource>
  </input>

  <output>
    <code>RES_FIRST_SITTING_HEIGHT</code>
    <measurementUnit>cm</measurementUnit>
    <dataType>DECIMAL</dataType>
    <captureMethod>MANUAL</captureMethod>
    <integrityChecks>
      <rangeCheck>
        <decimalMinValueMale>70.0</decimalMinValueMale>
        <decimalMaxValueMale>207.0</decimalMaxValueMale>
        <decimalMinValueFemale>70.0</decimalMinValueFemale>
        <decimalMaxValueFemale>207.0</decimalMaxValueFemale>
      </rangeCheck>
    </integrityChecks>
  </output>
```

```xml
<output>
  <code>RES_SEC_SITTING_HEIGHT</code>
  <measurementUnit>cm</measurementUnit>
  <dataType>DECIMAL</dataType>
  <captureMethod>MANUAL</captureMethod>
  <integrityChecks>
    <rangeCheck>
      <decimalMinValueMale>70.0</decimalMinValueMale>
      <decimalMaxValueMale>207.0</decimalMaxValueMale>
      <decimalMinValueFemale>70.0</decimalMinValueFemale>
      <decimalMaxValueFemale>207.0</decimalMaxValueFemale>
    </rangeCheck>
  </integrityChecks>
</output>

<output>
  <code>RES_THIRD_SITTING_HEIGHT</code>
  <measurementUnit>cm</measurementUnit>
  <dataType>DECIMAL</dataType>
  <captureMethod>MANUAL</captureMethod>
  <condition class="scriptDataSource">
    <valueType>boolean</valueType>
    <script>
      var diff =
$('SittingHeight:RES_FIRST_SITTING_HEIGHT').minus($('SittingHeight:RES_SEC
_SITTING_HEIGHT');
      diff.ge(0).value() ? diff.ge(1) : diff.le(-1);
    </script>
  </condition>
  <integrityChecks>
    <rangeCheck>
      <decimalMinValueMale>70.0</decimalMinValueMale>
      <decimalMaxValueMale>207.0</decimalMaxValueMale>
      <decimalMinValueFemale>70.0</decimalMinValueFemale>
      <decimalMaxValueFemale>207.0</decimalMaxValueFemale>
    </rangeCheck>
  </integrityChecks>
</output>

<output>
  <code>CALC_AVG_SITTING_HEIGHT</code>
  <measurementUnit>cm</measurementUnit>
  <dataType>DECIMAL</dataType>
  <captureMethod>COMPUTED</captureMethod>
  <dataSource class="scriptDataSource">
    <valueType>decimal</valueType>
    <unit>cm</unit>
    <script>

$('SittingHeight:RES_FIRST_SITTING_HEIGHT').plus($('SittingHeight:RES_SEC_
SITTING_HEIGHT')).div(2)
    </script>
```

```xml
        </dataSource>
    </output>

    <output>
      <code>CALC_DIFF_SITTING_HEIGHT</code>
      <measurementUnit>cm</measurementUnit>
      <dataType>DECIMAL</dataType>
      <captureMethod>COMPUTED</captureMethod>
      <dataSource class="scriptDataSource">
        <valueType>decimal</valueType>
        <unit>cm</unit>
        <script>

$('SittingHeight:CALC_AVG_SITTING_HEIGHT').minus($('SittingHeight:INPUT_ST
OOL_HEIGHT'))
        </script>
      </dataSource>
    </output>
```

```
        </instrumentParameters>
    </instrumentType>
```

Elements of an Instrument Descriptor File

These are the elements that can be included in an instrument descriptor file.

Elements	Description
`<instrumentType>`	Top-level element for defining an instrument and its attributes.
`<properties>`	Optional. Instruments have default configurations created during development (a path to a particular software, for example). These can be overridden using this node.
`<contraIndication>`	Contraindication, a condition that, if present, means the stage should not be carried out. If one or more contraindication elements are included in the instrument descriptor file, the first page of the stage will be about contraindications. The `<type>` can be: • `ASKED` - The interviewer will be prompted to ask the participant a question about this contraindication, and will have to select a radio button (Yes, No, or Doesn't Know) to register the participant's response. • `OBSERVED` - The interviewer will be prompted to observe whether or not a contraindication exists and will have to select the type from a drop-down list. If several contraindications are included, they may as well all be of the same type.
`<interpretative>`	Creates an interpretative variables used to capture information about the measurement. Like `<contraIndication>` elements, `<interpretative>` elements require a <type> (observed or asked). If one or more <interpretative> elements are included in the instrument descriptor file, a page about these variables will be included in the stage.
`<expectedMeasureCount>`	Optional. A data source can be used to set the minimum number of measurements required to validate the measurement. Can only be defined for instruments whose `<captureMethod>` is AUTOMATIC. Available since Onyx 1.3.0. If this element is not included, at least one measurement will be expected. See Allow Multiple Measures for details.
`<input>`	Defines an Input Parameter. An input parameter is a value that needs to be fed **to** the instrument in order to take the measurement. For example, height is an input parameter for a bioimpedence instrument. Input parameters are only required for electronic instruments with which Onyx will interface.
`<output>`	Defines an Output Parameter. An output parameter is a value that is captured **from** the instrument. These may be typed in by the operator or read directly from the instrument.
`<code>`	The code name of an instrument parameter. Must be unique within `<instrumentType>`. Serves as the key for the key-value pairs defined in the properties files for the stage. See Localizing a Physical Measurement Stage.
`<captureMethod>`	Defines how an output parameter is captured. MANUAL indicates that the operator is expected to provide the value. AUTOMATIC indicates that the value will be obtained by programmatic means. COMPUTED indicates that the value will be computed based on other values.
`<measurementUnit>`	Required for unit conversion when one parameter is used as input to another. All SI units (abbreviations) are supported.
`<dataType>`	Describes the type of data that is captured. Possible values for `<dataType>` are: BOOLEAN, DATA (DATA could be used to capture an image), DATE, DECIMAL, INTEGER, TEXT

`<mimeType>`	Describes the file type of output data. Should be defined for `<output>` whose `<dataType>` is `DATA`. Possible values for `<mimeType>` are: • application/vnd.ms-excel • text/xml • application/pdf • image/bmp • image/jpeg
`<dataSource>`	Instrument runtime value can be automatically obtained from an external data source.
`<condition>`	Optional. Can only be defined for output parameters. Uses a data source. Depending on how this condition is resolved, the operator may or may not be prompted to take another measurement. See `RES_THIRD_SITTING _HEIGHT` in instrument-descriptor.xml above.
`<repeatable>`	Optional. Can only be defined for output parameters. All the output parameters are expected to be assigned several values (i.e. repeatable) when a `<expectedMeasureCount>` is defined. If this is not the case (when an instrument returns a general test value within several trials values), a output parameter should be flagged as not being repeatable (use the `<repeatable>false</repeatable>` statement).

Using Data Sources in an Instrument Descriptor File

Instrument descriptor files can use a `<dataSource>` element for various purposes. One of the attributes of a `<dataSource>` element is `class`. Different type of datasources are available in Onyx (see data source specification page for more historical information). However, after recent improvements in Onyx, we recommend to use the 'scriptDataSource'.

The 'scriptDataSource' is made of:

- a `valueType` which can be one of: boolean, integer, decimal, text, binary, locale, date, datetime.
- a `script` statement using Magma javascript API.
- an optional boolean flag `sequence` specifying if the script returns a list of values (dfault is `false`).
- an optional `unit` attribute.

For example:

- Input parameters can specify a `<dataSource>` element. For example, when an instrument requires the participant's gender, an input parameter can use a `scriptDataSource` to fetch the value automatically from the corresponding variable:

```
<input>
  <code>Gender</code>
  <dataType>TEXT</dataType>
  <captureMethod>MANUAL</captureMethod>
  <dataSource class="scriptDataSource">
    <valueType>text</valueType>
    <script>$('Participants:Admin.Participant.gender')</script>
  </dataSource>
</input>
```

- A data source can be used to compute values (see `CALC_AVG_SITTING_HEIGHT` in instrument-descriptor.xml above).
- A data source can be used to define a condition for an output parameter (see `RES_THIRD_SITTING_HEIGHT` in instrument-descriptor.xml above).

Using Integrity Checks in an Instrument Descriptor File

Output parameter values can be checked for validity/integrity by using predefined `integrityChecks`:

Integrity Check	Description
`equalsValueCheck`	The output value must match a fixed value.

`equalsParameterCheck`	The output value must match another parameter's value. Useful for checking output parameter values against input parameter values.
`rangeCheck`	The output value must be within a certain male/female range of values. (see examples in instrument-descriptor.xml above)
`parameterSpreadCheck`	Checks that two parameter values are within a specified range; either absolute range (example: +/-1cm) or relative (example: +/-5%)

All integrity checks can be configured to produce a warning instead of an error by specifying its `type` element as `WARNING`.

Localizing a Physical Measurement Stage

After you have prepared the instrument-descriptor.xml file for a physical measurement stage, you must prepare properties files that contain localized strings for all the labels that will appear in the user interface. All elements in the instrument descriptor file that include a `<code>` element need to be localized. The `<code>` elements are the keys in the key-value pairs in the properties files. Here is a snippet from the English properties file for the blood pressure stage.

```
IV_ARM_USED=Arm to be used for measurement
IV_UPPER_ARM_LN=Length of upper arm
IV_UPPER_ARM_CIRC=Circumference of upper arm at midpoint
IV_ARMBAND_SIZE_USED=Armband size used
IV_S=Small
IV_M=Medium
IV_LG=Large
IV_XL=Extra Large
```

Adding a Physical Measurement Stage to Onyx

To add a physical measurement stage to your custom version of Onyx, you must:

1. Add the stage to the list of physical measurement stages.
2. Localize the name of the stage.
3. Set the position of the stage in the global stage ordering.

Adding the Stage to stages.xml

To add a physical measurement stage in your version of Onyx, you need to add a `<stage>` element to the `WEB-INF/config/jade/stages.xml` file.

You can define dependency conditions by using the `<stage>` elements for the onyx-demo stages as a model. See also Defining Stage Dependencies.

WEB-INF/config/jade/stages.xml

```
<stages>

   <stage>
     <name>BloodPressure</name>
     <module>jade</module>
     <stageDependencyCondition class="multipleCondition">
       <operator>AND</operator>
       <conditions>
         <variableCondition>
           <stageName>Consent</stageName>
           <variablePath>Consent.accepted</variablePath>
         </variableCondition>
         <variableCondition>
           <stageName>CIPreliminaryQuestionnaire</stageName>

<variablePath>CIPreliminaryQuestionnaire.QuestionnaireRun.timeEnd</variabl
ePath>
         </variableCondition>
         <variableCondition>
           <stageName>CIPreliminaryQuestionnaire</stageName>

<variablePath>CIPreliminaryQuestionnaire.STAGE_STATUS.BP_CI.active</variab
lePath>
           <operator>ne</operator>
           <data>
             <type>BOOLEAN</type>
             <value class="boolean">true</value>
           </data>
         </variableCondition>
       </conditions>
     </stageDependencyCondition>
   </stage>

[...]

</stages>
```

Localizing the Name of the Stage

To localize the name of the physical measurement in each available interface language, you edit each of the properties files in the `WEB-INF/con`
`fig/jade` directory .

For example, to localize the name of the BloodPressure stage:

- Add a line to the English properties file :

messages_en.properties

```
BloodPressure.description=Blood Pressure
```

- And a line to the French properties file:

---
messages_fr.properties

```
BloodPressure.description=Pression artérielle
```

Setting the Position of the Stage in the Interview

To establish where a stage fits in the interview, you add the stage name (for example: BloodPressure) to this line in the global Onyx configuration file :

WEB-INF/config/onyx-config.properties

```
\# A comma separated list of stage names that defines the ordering within
an Interview.
org.obiba.onyx.engine.staticStageOrder=Consent,HealthQuestionnaire,BloodPr
essure,...,Conclusion
```

Removing Traces of onyx-demo Stages

When you have finished defining your custom physical measurement stages, be sure to remove all traces of the stages copied from onyx-demo.

1. Remove unrequired `<stage>` elements from the `WEB-INF/config/jade/stages.xml` file.
2. Remove localized strings for the names of the unrequired stages from the properties files in `WEB-INF/config/jade`
3. Remove directories for unrequired stages from the `WEB-INF/config/jade/resources/instruments` directory.
4. Remove directories for unrequired stages from the `instruments` directory.

Defining Interview Reports

Overview

Onyx can print reports about the interview to give to participant. The Onyx page includes a **Print Reports** button. You must define the PDF files that will be printed.

- PDF Printer Name
- Creating Default Report
    - To create a PDF report
    - How to name report files
    - Where to put report files
    - How to get Onyx to complete form fields automatically
- Creating Custom Report

PDF Printer Name

The **Print Reports** button will be visible only if a PDF printer name has been declared. See Modifying the Printer section.

Creating Default Report

You should create a report in each of the languages that are required by your organization. At runtime, Onyx populates a dropdown list of the available report languages based on the report files that you have defined.

ℹ️ This procedure requires a single license for Acrobat Professional to prepare the report.

To create a PDF report

Repeat this procedure to create a PDF report for each language required by your organization.

1. Create a report in one of your required languages, using Microsoft Word or any tool of your choice.
2. Convert the form to PDF using a PDF converter.
3. Save the PDF using a name that includes the appropriate language code (and optionally, a country code). For example: `ParticipantReport_en.pdf` for the English version. See How to name report files below.
4. Edit the PDF using Adobe Acrobat Professional to include a text field wherever you want information to be supplied automatically by Onyx (for example, participant and interviewer information--see the table below).
    1. Set each field to read-only.
    2. Rename each field so it can be automatically completed by Onyx (see the table below; for example: Participant.fullName).
    3. Apply formatting to all fields that require it (using Acrobat field formatting; for example: date formatting).
    4. Select ADVANCED -> ENABLE USAGE RIGHTS IN ACROBAT READER in the top menu.
5. Save the PDF file.

How to name report files

The filenames of PDF reports must conform to this configurable naming scheme:

basename_language_country.pdf

where,

- *basename* is a configurable prefix for the filename (`ParticipantReport` by default). It can be configured in the `WEB-INF/config/jade/module-config.properties` file in your project
- *language* is the two-letter ISO Language Code (ISO-639-1) code for the form's language (example: `en` for English, `fr` for French). See http://www.loc.gov/standards/iso639-2/php/code_list.php.
- *country* (optional) is a two-letter ISO Country Code (ISO-3166). See http://www.iso.org/iso/english_country_names_and_code_elements.

Examples of valid report filenames:

- `ParticipantReport_en.pdf`

- `ParticipantReport_fr.pdf`

- `ParticipantReport_fr_CA.pdf`

Where to put report files

By default, the PDF files for reports should be located in the `WEB-INF/config/jade/resources` directory. Based on the report files in the `resources` directory, Onyx "discovers" the available languages and displays them at runtime. No further configuration is required.

You can change the `resources` directory by modifying this line in the `WEB-INF/config/jade/module-config.properties` file:

WEB-INF/config/jade/module-config.properties

```
org.obiba.onyx.jade.resources.path=${org.obiba.onyx.jade.path}/resources
```

How to get Onyx to complete form fields automatically

At runtime, Onyx can complete fields in your reports automatically. The mapping between fields in the PDF file for the report and Onyx variables is defined in `WEB-INF/config/jade/module-config.properties`. For a list of the Onyx variables that you can use in your report, see To view a list of Onyx variables.

Here is the field-to-variable mapping for the onyx-demo application:

```
# ...
org.obiba.onyx.jade.report.fieldToVariable=Participant\\.fullName=Onyx.Adm
in.Participant.fullName,
    Participant\\.age=Onyx.Admin.Participant.age,
    Participant\\.enrollmentId=Onyx.Admin.Participant.enrollmentId,
    Participant\\.barcode=Onyx.Admin.Participant.barcode,

BloodPressure\\.CALC_AVG_SYSTOLIC_BP=Onyx.BloodPressure.CALC_AVG_SYSTOLIC_
BP,

BloodPressure\\.CALC_AVG_DIASTOLIC_BP=Onyx.BloodPressure.CALC_AVG_DIASTOLI
C_BP,
    StandingHeight\\.CALC_AVG_HEIGHT=Onyx.StandingHeight.CALC_AVG_HEIGHT,
    Impedance310\\.RES_WEIGHT=Onyx.Impedance310.CALC_AVG_WEIGHT,
    Weight\\.RES_WEIGHT=Onyx.Weight.RES_WEIGHT,
    Impedance310\\.RES_BODY_MASS_INDEX=Onyx.Impedance310.CALC_AVG_BMI,
    Impedance310\\.RES_FAT_PERCENT=Onyx.Impedance310.CALC_AVG_FAT,
    Spirometry\\.RES_MEASURED_FEV1=Onyx.Spirometry.CALC_AVG_FEV1,
    Spirometry\\.RES_MEASURED_FVC=Onyx.Spirometry.CALC_AVG_FVC,
    Spirometry\\.RES_MEASURED_FEV1_FVC=Onyx.Spirometry.CALC_AVG_FEV1_FVC
```

Creating Custom Report

The reporting feature can be extended with additional custom reports. In order to achieve that, a report template is to be added to the configuration.

First, declare the custom PDF report template:

WEB-INF/spring/custom.xml

```
<beans ...>

  <...>

  <bean id="customParticipantReport"
class="org.obiba.onyx.print.impl.PdfTemplateReport" scope="session">
    <property name="name" value="${org.example.report.basename}" />
    <property name="label">
      <bean
class="org.springframework.context.support.DefaultMessageSourceResolvable">
        <constructor-arg value="CustomParticipantReportLabel" />
      </bean>
    </property>
    <property name="reportTemplateLoader">
      <bean class="org.obiba.onyx.core.io.support.LocalizedResourceLoader">
        <property name="resourcePath" value="${org.example.report.path}" />
        <property name="resourceName"
value="${org.example.report.basename}" />
        <property name="resourceExtension" value=".pdf" />
      </bean>
    </property>
    <property name="fieldToVariableMap"
value="${org.example.report.fieldToVariable}" />
  </bean>

</beans>
```

Some configuration properties have been declared in this XML snippet. These can be added to the main `onyx-config.properties`:

WEB-INF/config/onyx-config.properties

```
# Custom report
# Where the PDF templates are
org.example.report.path=${org.obiba.onyx.config.path}/reports
# The PDF template base name
org.example.report.basename=CustomParticipantReport
# The PDF template fields mapping to Onyx variables
org.example.report.fieldToVariable=Participant\\.fullName=Participants:Adm
in.Participant.fullName,Participant\\.age=Participants:Admin.Participant.a
ge,Participant\\.enrollmentId=Participants:Admin.Participant.enrollmentId,
Participant\\.barcode=Participants:Admin.Participant.barcode,BloodPressure
\\.CALC_AVG_SYSTOLIC_BP=BloodPressure:CALC_AVG_SYSTOLIC_BP,BloodPressure\\
.CALC_AVG_DIASTOLIC_BP=BloodPressure:CALC_AVG_DIASTOLIC_BP,StandingHeight\
\.CALC_AVG_HEIGHT=StandingHeight:CALC_AVG_HEIGHT,Impedance310\\.RES_WEIGHT
=Impedance310:CALC_AVG_WEIGHT,Weight\\.RES_WEIGHT=Weight:RES_WEIGHT,Impeda
nce310\\.RES_BODY_MASS_INDEX=Impedance310:CALC_AVG_BMI,Impedance310\\.RES_
FAT_PERCENT=Impedance310:CALC_AVG_FAT,Spirometry\\.RES_MEASURED_FEV1=Spiro
metry:CALC_AVG_FEV1,Spirometry\\.RES_MEASURED_FVC=Spirometry:CALC_AVG_FVC,
Spirometry\\.RES_MEASURED_FEV1_FVC=Spirometry:CALC_AVG_FEV1_FVC
```

The translation of the custom report label (defined by the `org.example.report.basename` property) can be added to the main `messages_*.properties` files.

The affected files are in this example:

```
WEB-INF
  config
      messages_en.properties
      messages_fr.properties
      onyx-config.properties
      reports
          CustomParticipantReport_en.pdf
          CustomParticipantReport_fr.pdf
  spring
      custom.xml
```

Configuring Sample Collection Stages

Overview

Onyx can include stages for collecting biological samples (blood, urine, and so on). This page explains how to configure a sample collection stage.

- Configuration Files for Sample Collection Stages
- Screenshot of a Sample Collection Stage
- What's Involved in Configuring a Sample Collection Stage
- Preparing a Bean for Your Sample Collection Stage
  - Naming the Stage
- Defining the Number of Samples Expected
- Defining the Barcode Structure
  - Using Predefined Barcode Parsers
  - Example Using Predefined Parsers

Configuration Files for Sample Collection Stages

The configuration files for the sample stage are found in the `$custom-onyx/WEB-INF/config/ruby` directory. SInce you copied the onyx-demo application, the directory should contain these standard configuration files:
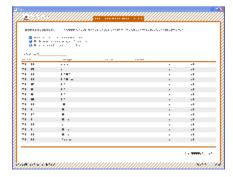
- `messages_en.properties` (contains English strings for all messages in all sample collection stage)
- `messages_fr.properties` (contains French strings for all messages in all sample collection stage)
- `module-context.xml` (contains tube registration configurations for all sample collection stages)
- `remarks.xml` (contains codes for a set of remarks that will be displayed in a drop-down list)
- `stages.xml` (contains definitions of all sample collection stages)

and these files specific to the onyx-demo application which includes two sample collection stages (Blood and Urine, and Urine only):

- `info-messages.bloodUrine.xml` (contains codes for custom messages in the Blood and Urine stage)
- `info-messages.urine.xml` (contains codes for custom messages in the Urine stage)

Screenshot of a Sample Collection Stage

The screenshot below shows the main page of a typical sample collection stage. The page includes a table of registered tubes. When the user scans the barcoce on a sample tube, the barcode is validated and if it passes validation, the tube is added to the table. Columns can be added to the table when you define the barcode structure.

What's Involved in Configuring a Sample Collection Stage

Configuring a sample collection stage involves the following:

- Defining the number of samples that the user is expected to collect
- Defining the structure of the barcode on sample labels and barcode validation
- Defining predefined remarks that the user choose from to describe the sample
- Defining custom information messages for the user (optional)
- Adding the stage to the `stages.xml` file
- Setting the stage order in Onyx

Preparing a Bean for Your Sample Collection Stage

The `module-context.xml` file that you copied from the onyx-demo application includes two bean definitions: one for each of the two sample collection stages defined in onyx-demo. You should copy one of these beans and modify its identifiers to fit your sample collection stage. For example, if your stage only for collecting blood, its identifiers would look like this:

```
                        WEB-INF/config/ruby/module-context.xml

  <util:map id="tubeRegistrationConfigurationMap">
    <entry key="BloodSamplesCollection"
        value-ref="bloodTubeRegistrationConfiguration" />
      [...]
  </util:map>


  <bean id="bloodTubeRegistrationConfiguration"
      class="org.obiba.onyx.ruby.core.domain.TubeRegistrationConfiguration">
    [...]
  </bean>
```

Naming the Stage

The string you use for `key` here is the name you will enter for the stage in the `stages.xml` file. See Adding the Stage to stages.xml.

Defining the Number of Samples Expected

To define the number of samples (tubes) that the user is expected to collect during the stage, edit this property in the bean for your stage in `WEB-INF/config/ruby/module-context.xml`:

```
<property name="expectedTubeCount" value="15" />
```

Defining the Barcode Structure

The barcode structure for the stage is defined in this property of the bean for your stage in `WEB-INF/config/ruby/module-context.xml`:

```
<property name="barcodeStructure">
```

A structured barcode is validated by an ordered list of **barcode part parsers**. Each part parser is responsible for validating a certain number of characters within the barcode. When valid, this portion is removed from the barcode and the remaining sequence of characters is passed on to the next part parser.

A barcode part parser can do the following:

- Notify the user with a specific message when a validation error occurs
- Validate the part based on any type of criteria, even data from within the database
- Optionally display a column (with localized labels) in the table of registered tubes that is displayed in Onyx. See the column labelled **Tube Type** in the Screenshot of a Sample Collection Stage.

Using Predefined Barcode Parsers

You can use these predefined parsers.

Parser Class	Description
FixedValueBarcodePartParser	Validates that the part is identical to a pre-defined value
AcceptableValuesBarcodePartParser	Validates that the part is identical to at least one element in a set of pre-defined values
ParticipantCodeBarcodePartParser	Validates that the part is identical to the current participant's code
RandomDigitsBarcodePartParser	Validates that the part is a sequence of digits
RegularExpressionBarcodePartParser	Validates that the part matches a specified regular expression

Example Using Predefined Parsers

For this barcode structure...

Study Code	Participant Code	Tube Type	Tube Sequence
11	54782	02	1

...the validation for the different parts of the barcode would be:

Barcode Part	Validation
Study Code	Must be a fixed value (`11`)
Participant Code	Must be the ID of participant currently being interviewed
Tube Type	Must be in a set of predefined values These values are also used as keys for localized strings that will be displayed in the table of registered tubes.
Sequence	Must be a digit

Here is how the above barcode structure would be validated:

WEB-INF/config/ruby/module-context.xml

```xml
<bean id="tubeRegistrationConfiguration"
class="org.obiba.onyx.ruby.core.domain.TubeRegistrationConfiguration">
   <constructor-arg value="${org.obiba.onyx.config.path}/ruby" />
   <property name="expectedTubeCount" value="16" />
   <property name="barcodeStructure">
     <bean class="org.obiba.onyx.ruby.core.domain.BarcodeStructure">
       <property name="parsers">
         <list>
           <!-- Study Code Parser -->
           <bean
class="org.obiba.onyx.ruby.core.domain.parser.impl.FixedValueBarcodePartPa
rser">
             <property name="fixedValue" value="11"/>
           </bean>
           <!-- Participant Code Parser -->
           <bean
class="org.obiba.onyx.ruby.core.domain.parser.impl.ParticipantCodeBarcodeP
artParser">
             <property name="activeInterviewService"
ref="activeInterviewService" />
             <property name="size" value="5" />
           </bean>
           <!-- Tube Type Parser -->
           <bean
class="org.obiba.onyx.ruby.core.domain.parser.impl.AcceptableValuesBarcode
PartParser">
             <!-- This parser will add a column to the table of registered
tubes -->
             <property name="partTitle">
               <bean
class="org.springframework.context.support.DefaultMessageSourceResolvable">
```

```xml
            <constructor-arg value="Ruby.BarcodePartColumn.TubeType" />
          </bean>
        </property>
        <property name="acceptableValues">
          <set>
            <value>01</value>
            <value>02</value>
            <value>03</value>
            <value>04</value>
            <value>05</value>
            <value>06</value>
            <value>07</value>
            <value>08</value>
            <value>09</value>
          </set>
        </property>
      </bean>
      <!-- Tube Sequence Parser -->
      <bean
class="org.obiba.onyx.ruby.core.domain.parser.impl.RandomDigitsBarcodePart
Parser">
        <!-- This parser will add a column to the table of registered
tubes -->
        <property name="partTitle">
          <bean
class="org.springframework.context.support.DefaultMessageSourceResolvable">
            <constructor-arg value="Ruby.BarcodePartColumn.Sequence" />
          </bean>
        </property>
        <property name="size" value="1"/>
      </bean>
    </list>
  </property>
</bean>
```

```
        </property>
    </bean>
```

This configuration would result in two columns being displayed in the table of registered tubes in Onyx.

Localizing Labels for Columns and Tube Types

The labels for columns and tube types must be added to the properties files in `WEB-INF/config/ruby`.

The labels associated with the values of barcode parts are looked up using a key composed of the part's value and the parser's column title, like this: `columnTitle.partValue`. If no label is found for a particular part value, the part's value itself becomes the label.

In the above example, the needed English labels would be defined by adding these lines to the English properties file:

WEB-INF/config/ruby/messages_en.properties
```
# Tube Type Column Title
Ruby.BarcodePartColumn.TubeType=Tube Type

# Tube Types Labels
Ruby.BarcodePartColumn.TubeType.01=P-Na-citrate
Ruby.BarcodePartColumn.TubeType.02=S-SST-LAB
Ruby.BarcodePartColumn.TubeType.03=P-PST
Ruby.BarcodePartColumn.TubeType.04=EDTA
Ruby.BarcodePartColumn.TubeType.05=EDTA-Hematol.
[...]

# Tube Sequence Column Title
Ruby.BarcodePartColumn.Sequence=Tube Sequence
```

Defining Custom Parsers

Part parsers classes must implement the IBarcodePartParser interface. As such, creating a custom parser is very straightforward.

Defining Remarks for Drop-down List

When a tube is added to the table of registered tubes (see screenshot), an Edit link is available. This Edit link leads opens a dialog that includes a drop-down list of remarks that the user can choose from to describe the sample. To populate this file, you must:

1. Define the codes for the strings that will populate the drop-down list in the `remarks.xml file`.
2. Define the corresponding localized strings in the `message_locale.properties` files.

Here are the codes for the remarks from the onyx-demo application:

WEB-INF/config/ruby/remarks.xml

```xml
<list>
    <remark>
        <code>TubeRegistration.Remark.Inappropriate_labeling</code>
    </remark>
    <remark>
        <code>TubeRegistration.Remark.Hemolyzed_sampling</code>
    </remark>
    <remark>
        <code>TubeRegistration.Remark.Clotted_sample</code>
    </remark>
    <remark>
        <code>TubeRegistration.Remark.Lipemic_sample</code>
    </remark>
    <remark>
        <code>TubeRegistration.Remark.Insufficient_quantity</code>
    </remark>
    <remark>
        <code>TubeRegistration.Remark.Inappropriate_tube</code>
    </remark>
    <remark>
        <code>TubeRegistration.Remark.Expired_tube</code>
    </remark>
    <remark>
        <code>TubeRegistration.Remark.Blood_in_urine</code>
    </remark>
</list>
```

Here are the English strings for the remarks in the onyx-demo application:

WEB-INF/config/ruby/messages_en.properties

```
# Remarks
TubeRegistration.Remark.Inappropriate_labeling=Inappropriate labeling
TubeRegistration.Remark.Hemolyzed_sampling=Hemolyzed sampling
TubeRegistration.Remark.Clotted_sample=Clotted sample
TubeRegistration.Remark.Lipemic_sample=Lipemic sample
TubeRegistration.Remark.Insufficient_quantity=Insufficient quantity
TubeRegistration.Remark.Inappropriate_tube=Inappropriate tube
TubeRegistration.Remark.Expired_tube=Expired tube
TubeRegistration.Remark.Blood_in_urine=Blood in urine
```

Adding Custom Messages

As shown in the screenshot, a sample collection stage can display a number of messages near the top of the main page of a sample collection stage.

Defining Custom Messages for One Sample Collection Stage

To set up custom messages for a particular sample collection stage, you must:

1. Create a file containing the codes for the stage's messages . For example, the onyx-demo application, has two such files: one for each of its sample collection stages (`info-messages.bloodUrine.xml` and `info-messages.urine.xml`). Put the file in the `WEB-INF/config/ruby` directory.
2. Add localized strings for the custom messages to the `messages_locale.properties` files in `WEB-INF/config/ruby` directory.
3. Edit the `WEB-INF/config/ruby/module-context.xml` file and add a `<property name="infoMessagesFile">` element to the bean definition for the stage. Set the `value` to the   appropriate filename. For example:

---

src/main/webapp/WEB-INF/config/ruby/module-context.xml

```
<bean id="bloodUrineTubeRegistrationConfiguration"
class="org.obiba.onyx.ruby.core.domain.TubeRegistrationConfiguration">
    <constructor-arg value="${org.obiba.onyx.config.path}/ruby" />


    <property name="infoMessagesFile" value="info-messages.bloodUrine.xml"
/>


    (snip)
  </bean>


  <bean id="urineTubeRegistrationConfiguration"
class="org.obiba.onyx.ruby.core.domain.TubeRegistrationConfiguration">
    <constructor-arg value="${org.obiba.onyx.config.path}/ruby" />


    <property name="infoMessagesFile" value="info-messages.urine.xml" />


    (snip)
  </bean>
```

Using the Same Messages for All Sample Collection Stages

1. Create a file named `info-messages.xml` that contains the codes for the messages.
2. Add localized strings for the messages to the `messages_locale.properties` files in `WEB-INF/config/ruby` directory.

NOTE: Do not add any `<property name="infoMessagesFile">` elements to the `module-context.xml` file.

Adding a Sample Collection Stage to Onyx

To add a sample collection stage to your custom version of Onyx, you must:

1. Add the stage to the list of sample collection stages.
2. Localize the name of the stage.
3. Set the position of the stage in the global stage ordering.

Adding the Stage to stages.xml

To add a sample collection to your version of Onyx, you need to add a `<stage>` element to the `WEB-INF/config/ruby/stages.xml` file. You can define any dependency conditions by using the `<stage>` elements for the onyx-demo stages as a model. See also Defining Stage Dependencies.

The `<name>` of the stage must be the name used in the module_context file. See Naming the Stage.

WEB-INF/config/ruby/stages.xml

```xml
<stages>

  <stage>
    <name>BloodUrineSamplesCollection</name>
    <module>ruby</module>
    <stageDependencyCondition class="multipleCondition">
      <operator>AND</operator>
      <conditions>
        <variableCondition>
          <stageName>Consent</stageName>
          <variablePath>Consent.accepted</variablePath>
        </variableCondition>
        <variableCondition>
          <stageName>CIPreliminaryQuestionnaire</stageName>

<variablePath>CIPreliminaryQuestionnaire.QuestionnaireRun.timeEnd</variabl
ePath>
        </variableCondition>
        <variableCondition>
          <stageName>CIPreliminaryQuestionnaire</stageName>

<variablePath>CIPreliminaryQuestionnaire.STAGE_STATUS.BSC_CI.active</varia
blePath>
          <operator>ne</operator>
          <data>
            <type>BOOLEAN</type>
            <value class="boolean">true</value>
          </data>
        </variableCondition>
      </conditions>
    </stageDependencyCondition>
  </stage>

[...]

</stages>
```

Localizing the Name of the Stage

To localize the name of the physical measurement in each available interface language, you edit each of the properties files in the `WEB-INF/config/ruby` directory .

For example, to localize the name of the BloodPressure stage:

- Add a line to the English properties file:

WEB-INF/config/ruby/messages_en.properties

```
BloodUrineSamplesCollection.description=Blood/Urine Samples Collection
```

- And a line to the French properties file:

WEB-INF/config/ruby/messages_en.properties

```
BloodUrineSamplesCollection.description=v
```

Setting the Position of the Stage in the Interview

To establish where a stage fits in the interview, you add the stage name (for example: BloodSamplesCollection) to this line in the global Onyx configuration file :

WEB-INF/config/onyx-config.properties

```
# A comma separated list of stage names that defines the ordering within an
Interview.
org.obiba.onyx.engine.staticStageOrder=Consent,HealthQuestionnaire,BloodSa
mplesCollection,...,Conclusion
```

Removing Traces of onyx-demo Stages

When you have finished defining your custom sample collection stages, be sure to remove all traces of the stages copied from onyx-demo.

1. Remove onyx-demo `<stage>` elements from the `WEB-INF/config/ruby/stages.xml` file.
2. Remove onyx-demo <entry> and <bean> elements from the `WEB-INF/config/ruby/module-context.xml` file.
3. Remove the localized strings for the onyx-demo stages from the properties files in `WEB-INF/config/ruby`. Check these sections:
   Stage Descriptions
   Info Messages for BloodUrine
   Info Messages for Urine
4. Remove files for onyx-demo custom messages from the `WEB-INF/config/ruby` directory:
   info-messages.bloodUrine.xml
   info-messages.urine.xml

Configuring Logs for Experimental Conditions and Instrument Calibration

Overview

Logs for recording experimental conditions and instrument calibrations are an optional feature of Onyx that users manage from the Workstation page of Onyx.

Experimental condition logs allow periodic capture of environmental conditions such as room temperature and relative humidity. Similarly, instrument calibration log may be done at any time to measure the accuracy of the instruments used in the study.

- The Configuration File
- Customizing the Configuration File
- Localizing the Log Interface
- If You Don't Need Logs

The Configuration File

One configuration file (`experimental-conditions.xml`) is used to set up both experimental condition logs and instrument calibration logs. By copying the onyx-demo application, you should have a copy of its `experimental-conditions.xml` at the following location:

```
WEB-INF/config/jade/experimental-conditions.xml
```

The following example contains one experimental condition log and one instrument calibration log.

WEB-INF/config/jade/experimental-conditions.xml

```
<?xml version="1.0" encoding="UTF-8"?>
```

```xml
<list>
  <experimentalConditionLog>
    <name>RoomCharacteristics</name>
    <attribute>
      <name>Temperature</name>
      <type>INTEGER</type>
      <unit>°C</unit>
      <validators>
        <dataValidator dataType="INTEGER">
          <rangeValidator>
            <minimum>-40</minimum>
            <maximum>35</maximum>
          </rangeValidator>
        </dataValidator>
      </validators>
    </attribute>
    <attribute>
      <name>RelativeHumidity</name>
      <type>INTEGER</type>
      <unit>%</unit>
      <validators>
        <dataValidator dataType="INTEGER">
          <rangeValidator>
            <minimum>0</minimum>
            <maximum>100</maximum>
          </rangeValidator>
        </dataValidator>
      </validators>
    </attribute>
  </experimentalConditionLog>

  <instrumentCalibration>
    <name>StadiometerFullAlignmentCalibrationLog</name>
    <instructions>
      <string>StadiometerFullAlignmentCalibrationLog.Instruction1</string>
      <string>StadiometerFullAlignmentCalibrationLog.Instruction2</string>
    </instructions>
    <instrumentType>StandingHeight</instrumentType>
    <attribute>
      <name>Alignment50to70cm</name>
      <type>TEXT</type>
      <allowedValues>
        <string>Yes</string>
        <string>No</string>
      </allowedValues>
    </attribute>
    <attribute>
      <name>Alignment100to120cm</name>
      <type>TEXT</type>
      <allowedValues>
        <string>Yes</string>
        <string>No</string>
      </allowedValues>
```

```
          </attribute>
          <attribute>
            <name>Alignment150to170cm</name>
            <type>TEXT</type>
            <allowedValues>
              <string>Yes</string>
              <string>No</string>
            </allowedValues>
          </attribute>
          <attribute>
            <name>HeightOfMetreStick</name>
            <type>INTEGER</type>
            <unit>mm</unit>
            <validators>
              <dataValidator dataType="INTEGER">
                <rangeValidator>
                  <minimum>900</minimum>
                  <maximum>1100</maximum>
                </rangeValidator>
              </dataValidator>
            </validators>
          </attribute>
        </instrumentCalibration>

  </list>
```

Customizing the Configuration File

If your organization requires experimental condition logs and/or instrument calibrations, modify the configuration file copied from onyx-demo to suit your organization needs:

```
WEB-INF/config/jade/experimental-conditions.xml
```

Localizing the Log Interface

Users will calibrate instruments and make logs entries from the Workstation page of the Onyx user interface. You must localize the strings that are displayed by editing the properties files that are in the same directory as `experimental-conditions.xml` file. By default

Here is an example of the English properties file. (Note the "en" in file name.) The French properties file name would be `src/main/webapp/WEB-INF/config/jade/messages_fr.properties`.

```
RoomCharacteristics=Room Characteristics
RelativeHumidity=Relative Humidity
Temperature=Temperature

RoomCharacteristicsLog=Room Characteristics Log
RelativeHumidityLog=Relative Humidity Log
TemperatureLog=Temperature Log

StadiometerFullAlignmentCalibrationLog=Stadiometer: Full Alignment
Calibration Log
StadiometerFullAlignmentCalibrationLog.Instruction1=Place the tape measure
"joints" of the stadiometer and record if the "mm" marks are aligned.
StadiometerFullAlignmentCalibrationLog.Instruction2=Measure the height of
the meter stick, ensuring it is vertical.
Alignment50to70cm=Alignment 50 - 70 cm
Alignment100to120cm=Alignment 100 - 120 cm
Alignment150to170cm=Alignment 150 - 170 cm
HeightOfMetreStick=Measured height of meter stick
```

If You Don't Need Logs

If your organization does not intend to keep logs of experimental condition logs or instrument calibrations, simply delete the configuration file:

```
WEB-INF/config/jade/experimental-conditions.xml
```

Configuring Data Export and Purge

Overview

Onyx data consists of participant data, workstation data (experimental condition data), and instrument data (calibration data). All three kinds of data can be exported to zip files. Only participant data can be purged (deleted from the Onyx database)

- Configuring Data Export
    - Specifying the Export Options
    - Encrypting Data Upon Export
        - To increase the keySize
    - Modifying the Location of the Keystore
    - Modifying the Export Directory
    - Automating Export
- Configuring Purge of Participant Data
    - Default purge configuration
    - Modifying the default purge configuration
    - Allow Multiple Interviews of a Participant
    - Automating Purge

Configuring Data Export

Three types of data can be exported:

- Participant data (interview data)
- Workstation data (data from experimental condition logs)
- Instrument data (calibration data)

You configure export of all three types of data by editing the `export-destinations.xml` file in the `WEB-INF/config` directory.

An entry for each stage associated with a Participant must be added to the `export-destinations.xml` file if the data from that stage is to be

exported. A stage can be a questionnaire or a physical measure such as Standing Height.

Data can be exported to multiple destinations. The `export-destinations.xml` file contains a `<destination>` element for each destination, and each `<destination>` element can contain the following:

- a `<name>` element that must be unique within the set of destinations
- an optional `<options>` element to specify the format of the destination (CSV, XML files or remote Opal server). See details in #Specifying the Export Options.
- an optional `<encrypt>` element to specify if and how encryption of the output file should occur. If a `<destination>` does not include an `<encrypt>` element, data exported to that destination will not be encrypted. See details in #Encrypting Data Upon Export.
- a `<valueset>` element for each type of data (now known as an `entity` in the subterranean world of Magma) that will be exported to the destination; a `<valueset>` element, in turn, contains:
    - a `<entities>` element that defines which entities to include in or exclude from the export,
    - a `<variables>` element that defines which variables to include in or exclude from the export. Variables can be included/excluded by a name `prefix` or a `match` regular expression pattern.

The example below (taken from the `export-destinations.xml` file for the onyx-example application) can serve as a starting point for your configuring your export destinations. For information about how to write the script in the `<javascript>` element, see Magma Javascript API.

As you can see in the example, one of the conditions that can be used in a `<script>` for participant data is the status of the participant interview. Four interview statuses are possible: COMPLETED, CANCELLED, CLOSED, IN_PROGRESS.

WEB-INF/config/export-destinations.xml

```
<destinations>

  <!-- DCC destination: anonymous Participant data -->
  <destination>
    <name>DCC</name>

    <!-- Export in CSV with provided options -->
    <options>
      <format>CSV</format>
      <characterSet>ISO-8859-1</characterSet>
      <separator>,</separator>
      <quote>"</quote>
    </options>

    <!-- Do not export variables identifying participant -->
    <valueset entityType="Participant" valueTable="Participants">
      <entities>
        <excludeAll />
        <script type="INCLUDE">
          <javascript><![CDATA[// Include any ValueSet that has 'CLOSED' or
'COMPLETED' as a value for the 'Participant.Interview.Status' variable

$('Participants:Admin.Interview.status').any('CLOSED','COMPLETED')]]></jav
ascript>
        </script>
        <script type="EXCLUDE">

<javascript><![CDATA[$('Participants:Admin.Interview.exportLog.destination
').any('DCC.Participants')]]></javascript>
        </script>
      </entities>
      <variables>
        <variableName type="EXCLUDE" prefix="Admin.Participant" />
        <variableName type="INCLUDE" prefix="Admin.Participant.barcode" />
        <variableName type="INCLUDE"
```

```xml
          prefix="Admin.Participant.PostalCodePrefix" />
        </variables>
      </valueset>


      <!-- For each stage specify which participants data are to be exported
-->
      <valueset entityType="Participant" valueTable="ArtStiffness">
        <entities>
          <excludeAll />
          <script type="INCLUDE">
            <javascript><![CDATA[// Include any ValueSet that has 'CLOSED' or
'COMPLETED' as a value for the 'Participant.Interview.Status' variable

$('Participants:Admin.Interview.status').any('CLOSED','COMPLETED')]]></jav
ascript>
          </script>
          <script type="EXCLUDE">

<javascript><![CDATA[$('Participants:Admin.Interview.exportLog.destination
').any('DCC.ArtStiffness')]]></javascript>
          </script>
        </entities>
      </valueset>
      <!-- More stages ... ->

      <!-- Export experimental conditions -->
      <valueset entityType="Workstation" valueTable="Workstations">
        <entities>
          <script type="EXCLUDE">

<javascript><![CDATA[$('Workstation.exportLog.destination').any('DCC.Works
tations')]]></javascript>
          </script>
        </entities>
      </valueset>

      <!-- Export instrument calibrations -->
      <valueset entityType="Instrument" valueTable="Instruments">
        <entities>
          <script type="EXCLUDE">

<javascript><![CDATA[$('Instrument.exportLog.destination').any('DCC.Instru
ments')]]></javascript>
          </script>
        </entities>
      </valueset>

    </destination>

    <!-- Appointment Unit destination: nominative Participant data -->
    <destination>
      <name>Appointment Unit</name>
```

```xml
    <!-- Default format is XML. 'encrypt' section allows encryption of the
file. -->
    <encrypt>
      <algorithm>DESede</algorithm>
      <mode>CBC</mode>
      <padding>PKCS5Padding</padding>
      <keySize>112</keySize>
    </encrypt>


    <!-- -->
    <valueset entityType="Participant" valueTable="Participants">
      <entities>
        <excludeAll />
        <script type="INCLUDE">
          <javascript><![CDATA[// Include any ValueSet that has 'CLOSED' or
'COMPLETED' as a value for the 'Participant.Interview.Status' variable

$('Participants:Admin.Interview.status').any('CLOSED','COMPLETED')]]></jav
ascript>
        </script>
        <script type="EXCLUDE">

<javascript><![CDATA[$('Participants:Admin.Interview.exportLog.destination
').any('Appointment Unit.Participants')]]></javascript>
        </script>
      </entities>
      <variables>
        <excludeAll />
        <variableName type="INCLUDE" prefix="Admin.onyxVersion" />
      </variables>
    </valueset>

    <valueset entityType="Participant" valueTable="Consent">
      <entities>
        <excludeAll />
        <script type="INCLUDE">
          <javascript><![CDATA[// Include any ValueSet that has 'CLOSED' or
'COMPLETED' as a value for the 'Participant.Interview.Status' variable

$('Participants:Admin.Interview.status').any('CLOSED','COMPLETED')]]></jav
ascript>
        </script>
        <script type="EXCLUDE">

<javascript><![CDATA[$('Participants:Admin.Interview.exportLog.destination
').any('Appointment Unit.Consent')]]></javascript>
        </script>
      </entities>
      <variables>
        <variableName type="EXCLUDE" prefix="actions" />
      </variables>
    </valueset>
```

```
   </destination>
```

```
    </destinations>
```

Specifying the Export Options

The `<options>` element is optional, but when specified, it can contain some parameters that specify the export format, participant ID to be used etc.

Option	Description
`<format>`	Possible values are: • CSV • XML (default) • OPAL : export by connecting directly to a Opal server
`<useEnrollmentId>`	Use enrolment ID instead of the Participant interview ID (default is false). Allowed values are: • true • false
`<copyNullValues>`	Export Participant's empty value sets (default is false). Allowed values are: • true • false
`<characterSet>`	CSV character set to be used (default is system's one).
`<separator>`	CSV separator character (default is ,).
`<quote>`	CSV quote character (default is ").
`<opalDatasource>`	Name of the datasource in Opal to which data will be exported.

By default (i.e. when `<options>` element is not specified), the format is `XML`, participant interview ID is used and value sets with only null values are not exported.

Encrypting Data Upon Export

The `<encrypt>` element is optional, but when specified, it can contain some parameters that affect how the encryption occurs. Four parameters can be overridden: `algorithm`, `mode`, `padding` and `keySize`. Without defining what these mean exactly, here are the possible combinations:

Algorithm	Mode	Padding	Maximum Key Size
AES	CBC	NoPadding	128
AES	CBC	PKCS5Padding	128
AES	ECB	NoPadding	128
AES	ECB	PKCS5Padding	128
DES	CBC	NoPadding	56
DES	CBC	PKCS5Padding	56

DES	ECB	NoPadding	56
DES	ECB	PKCS5Padding	56
DESede	CBC	NoPadding	168
DESede	CBC	PKCS5Padding	168
DESede	ECB	NoPadding	168
DESede	ECB	PKCS5Padding	168
RSA	ECB	PKCS1Padding	2048
RSA	ECB	OAEPPadding	2048

If you use an empty `<encrypt/>` element (do not specify any parameters), these default values will be used:

algorithm	mode	padding	keySize
AES	CBC	NoPadding	128

To increase the keySize

Note that the maximum keySize can be extended by installing the `Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files`. You can get these files on the [Java security page](#) and install the files following their directions.

Modifying the Location of the Keystore

The onyx keystore is required for encrypting participant data upon export. The location of the keystore file can be configured in the `onyx-config.properties` file.

Property Key	Description
org.obiba.onyx.keystore.file	The path and name of the keystore file

Modifying the Export Directory

By default, the demo configuration writes export files to a directory named `target` under the "current working directory" (this is usually Tomcat's home directory).

You can change this output directory by editing this property in the `onyx-config.properties` file:

Property Key	Description
org.obiba.onyx.export.path	The path to export data to

Automating Export

Onyx exposes a web service for automating the export action. Access to this web service requires authentication and authorization (user must have Participant Manager role).

Example using [cURL](#) command line:

```
# Replace username, password and onyx base url
curl -XPOST -H "Authorization: Basic `echo -n "username:password" |
base64`" http://localhost:8080/ws/export
```

Configuring Purge of Participant Data

Participants and their data can be purged from the Onyx database. This is typically done at some point after the data has been export. Data that is selected for the purge is permanently removed from the Onyx database. The purge function is configured in a file called `purge.xml` in the `WEB-INF/config` directory.

Default purge configuration

By copying the onyx-demo application, you will have the following default version of the `purge.xml` file. Notice that certain elements in `purge.xml` resemble those in `export-destinations.xml`. Both purge and export involve defining which data to exclude and which data to include in the operation, and javascripts are used to fine-tune what is included (and sometimes what is excluded).

WEB-INF/config/purge.xml

```
<list>
  <purge>
    <valueset entityType="Participant" valueTable="Participants">
      <entities>

        <excludeAll />

        <script type="INCLUDE">
          <javascript>
              // If the interview has been exported
              // AND the interview was finished more than
              // 'org.obiba.onyx.participant.purge' days ago,
              // then purge.
              ($('Admin.Interview.exportLog.destination').empty().not()).
                  and($('Admin.Interview.endDate').isNull().not()).
                  and(now().after($('Admin.Interview.endDate').
                  add(onyx('org.obiba.onyx.participant.purge'))))
          </javascript>
        </script>

        <script type="INCLUDE">
          <javascript>
              // If the interview has been 'CLOSED' or 'CANCELLED'
              // AND the interview was finished
              // before the last export, then purge.
              ($('Admin.Interview.endDate').isNull().not()).
                  and(onyx('org.obiba.onyx.lastExportDate').
                  after($('Admin.Interview.endDate'))).
                  and($('Admin.Interview.status').
                  any('CLOSED','CANCELLED'))
          </javascript>
        </script>

      </entities>
    </valueset>
  </purge>
</list>
```

In the default version of `purge.xml`:

- The first `<script type="INCLUDE">` element selects data for purging based on dates. The property `org.obiba.onyx.participant.purge` which is defined in `onyx-config.properties` allows you to set a number of days. Data older than the configured number of days will be purged. The default number of days is 0.
- The second `<script type="INCLUDE">` element selects data for purging based on interview status. If the interview status is `CLOSED` or `CANCELED`, and the interview was finished prior to the last export, then purge.

Modifying the default purge configuration

You can configure purge of participant data as follows:

- If your organization wants to customize the purge function rather than use the default configuration, you can modify the `<script>` elements in purge.xml. For information about how to write the script in the `<javascript>` element, see Magma Javascript API. You can also contact the Onyx development team to discuss how to fine-tune the javascripts.

- If your organization wants to use the default configuration of the purge function, edit the following line in the `onyx-config.properties` file to define a "purge before" date:

---

onyx-config.properties

```
# Number of days used to select data for purging (data older than this
number of days will be purged)
org.obiba.onyx.participant.purge=15
```

Allow Multiple Interviews of a Participant

By default Onyx does not allow a Participant, identified by its enrolment ID, to be interviewed several times on the same Onyx server instance, even after participant's data were purged. In case of Onyx is configured so that walk-in participants can be enrolled, Onyx has no way to ensure that the same person volunteers several times (no enrolment ID is available and no personal data are kept after purge).

The following line can be added in the `onyx-config.properties` file to support several receptions of an enrolled Participant (interviews will be consecutive to a data purge).

---

onyx-config.properties

```
# Allow multiple interviews for a participant (after participant data has
being purged). Participants will be
# identified by their enrollment ID. Multiple interviews will require
different participant interview ID to be assigned.
org.obiba.onyx.participant.purge.multipleInterview=true
```

Automating Purge

Onyx exposes a web service for automating the purge action. Access to this web service requires authentication and authorization (user must have System Administrator role).

Example using cURL command line:

```
# Replace username, password and onyx base url
curl -XPOST -H "Authorization: Basic `echo -n "username:password" |
base64`" http://localhost:8080/ws/purge
```

Generating a Custom War File

Overview

When you have configured and customized the components of Onyx to your satisfaction, generate the war file for your custom version of Onyx as follows:

1. Edit the `pom.xml` file that generates the war file. It should be in the {*custom-demo*} directory that you created when you copying the onxy-demo source files. The snippet below gives an indication of its content.
2. Change the values of the `<name>`, `<description>`, `<groupId>`, `<artifactId>` and `<finalName>` elements. Use values appropriate for your custom version of Onyx.

<table>
<tr><td colspan="1" align="center">**custom-onyx/pom.xml**</td></tr>
</table>

```
[...]


    <name>Example Onyx Deployment</name>
    <description>Example Assessment Center Web Application
Deployment</description>

    <groupId>org.obiba.onyx</groupId>
    <artifactId>onyx-demo</artifactId>
    <packaging>war</packaging>

[...]


    <finalName>onyx-demo</finalName>

[...]
```

3. From the project directory (*custom-onyx*), run this Maven command to generate the war file:

```
mvn package
```

The result of running this command will be a target directory that includes a war file named as per the value you entered in the <finalName> element. This is the war file that will be used to deploy your custom version of Onyx on the server - save it somewhere. You can delete the rest of the contents of the target directory as well as the target directory itself once you have copied the war file elsewhere.

Testing the War File

Test that Tomcat can use the war file to deploy your custom version of Onyx as follows:

1. Stop Tomcat.
2. Put the war file in the Tomcat `webapps` directory.
3. Restart Tomcat.
4. Check that you can access your version of Onyx by pointing your browser to http://localhost:8080/*custom-onyx*.

Onyx Installation Guide

Contents of this Guide

- Introduction
- What You Need from Your Organization
- Hardware Requirements
- Software Requirements
- Setting Up the Server
- Setting Up the Workstations

Introduction

Onyx is a web application that runs on a server. Users access it on client workstations via a browser over a local network.

This guide is for whoever will set up Onyx in an assessment centre--typically, a system administrator.

The guide covers hardware and software requirements and includes procedures for deploying Onyx on a server and setting up the client workstations.

What You Need from Your Organization

Your organization should provide you with the following:

- A war file (*custom-onyx*.war) that contains a version of Onyx customized to your organization's requirements
- An appointment list (may be an Excel or XML file)

> The war file is actually a zip file. You will need some information from two files in the war file before they are extracted during installation. Your organization may provide you with copies of the files. Or you can extract copies of the two files from the war file:
>
> - onyx-config.properties
> - export-destinations.xml

Hardware Requirements

Server Hardware Requirements

Component	Requirement
CPU	Recent server-grade or high-end consumer-grade processor
Memory (RAM)	Minimum: 2 GB
Disk space	Minimum: 160 GB Rule of thumb calculation: 10 GB for operating system + ((8 GB/2000 participants) * (expected number of participants))

Workstation Hardware Requirements

Component	Requirement
CPU	Mid to high-end consumer-grade processor
Memory (RAM)	Minimum: 1 GB
Disk space	Enough to satisfy the requirements of any proprietary software that comes with instruments that will be connected to and launched by Onyx. Onyx does not have any storage requirements for client workstations.
USB ports	Minimum: 2 (if barcode scanner and electronic signature pad will be used); as many as possible Number of additional ports required depends on the number of instruments that will be connected to the workstation.
COM ports	Depends on number of instruments that will be connected
Display	Minimum resolution: 1024 x 768 If using touchscreens: touchscreen monitors should have a sturdy base so they are stable during use.

Software Requirements

Server Software Requirements

You must install the following software on the Onyx server.

Software	Suggested Version	Download Link	Use	Installation/Configuration

Firefox	Latest	http://www.mozilla-europe.org/en/firefox	Browser- useful for checking Onyx	
JRE	JRE 1.6.x	http://www.java.com/en/download/manual.jsp or http://java.sun.com/javase/downloads/index.jsp	Java runtime environment - needed to run Tomcat and Onyx	http://java.sun.com/javase/6/webnotes/install/index.html
Tomcat	6.0.x	http://tomcat.apache.org/download-60.cgi	Servlet container - needed to run Onyx	Fine-tuning Tomcat
MySQL	5.0.x	http://www.mysql.com/downloads/mysql/	Database management system - stores data collected using Onyx	Configuring MySQL
Custom war file for Onyx (`custom-onyx .war`)		Get it from whoever customized Onyx for your organization	Contains all Onyx files	Deploying Onyx
JCE (if required by your organization)	Latest	http://java.sun.com/javase/downloads/index.jsp	Java Cryptography Extension. In most cases, Onyx's default handling of encryption should be sufficient. Only install JCE if your organization has specified that it should be installed.	Copy the jar files from the downloaded jce zip file to this directory: `JAVA_HOME\Java\jre6\lib\security` which is usually: `C:\Program Files\Java\jre6\lib\security`

Workstation Software Requirements

You must install the following software on each Onyx workstation.

Software	Suggested Version	Download Link	Use	Installation/Configuration
Firefox	Latest	http://www.mozilla-europe.org/en/firefox	Needed to import an SSL certificate.	
Acrobat Reader	Latest	http://get.adobe.com/uk/reader/	Browser plug-in - needed to display electronic consent forms and participant reports (PDF files).	

Setting Up the Server

Overview

Setting up the Onyx server involves some or all of the following tasks:

- Fine-tuning Tomcat
  - General Comments about Tomcat Memory Settings
  - Using Tomcat on Ubuntu
    - To allocate memory on Ubuntu
    - To disable the security manager on Ubuntu
  - Using Tomcat as a Windows Service
    - To allocate memory on Windows
    - To enable printing from Tomcat
- Configuring MySQL
  - To check the database connection settings
  - To create a database for Onyx in MySQL
- Setting Up an SSL Connection
  - To create a keystore for the SSL connection

Fine-tuning Tomcat

For general tips related to Tomcat, see http://tomcat.apache.org/.

> The following tips assume that you are using Sun's Java Virtual Machine

General Comments about Tomcat Memory Settings

On most default Tomcat installations, the memory allocated to the service is insufficient. You must increase the memory allocated to Tomcat by tweaking two memory settings.

- `-Xmx`
- `-XX:MaxPermSize`

In the OS-specific procedures below (see Allocating Memory on Ubuntu and Allocating Memory on Windows), we assume that your server has 2GB of RAM. We set `-Xmx` to 1024 and `-XX:MaxPermSize` to 256m.

- If your server has more than 2GB of RAM, you can increase the `-Xmx` setting. It is not necessary to increase the `-XX:MaxPermSize` setting.
- If your server has less than 2GB of RAM, you can decrease the `-Xmx` and the `-XX:MaxPermSize` settings. Try to keep the same ratio (4:1). **Do not** allocate less than 128m to `-XX:MaxPermSize`.

Using Tomcat on Ubuntu

Two issues need to be addressed when running Tomcat on Ubuntu: insufficient memory and the security manager.

To allocate memory on Ubuntu

To allocate memory to Tomcat, you need to create an environment variable. If Tomcat was installed using `apt`, you must edit the `/etc/default /tomcat6` file as shown below.

/etc/default/tomcat6

```
JAVA_OPTS="-Xmx1024M -XX:MaxPermSize=256M"
```

To disable the security manager on Ubuntu

Comment out the `#TOMCAT6_SECURITY=yes` and add the `TOMCAT6_SECURITY=no` line as shown below.

```
# Use the Java security manager? (yes/no, default: yes)
# WARNING: Do not disable the security manager unless you understand
# the consequences!
#TOMCAT6_SECURITY=yes
TOMCAT6_SECURITY=no
```

Using Tomcat as a Windows Service

Two issues need to be addressed when running Tomcat as a service on Windows: insufficient memory and making the printer available to Tomcat.

To allocate memory on Windows

1. Open the **Apache Tomcat Properties** dialog in either of these ways:
    * If you have a Tomcat icon on your taskbar, right-click it and select **Configure**.
    * From the **Start** menu, select **All Programs** > **Apache Tomcat 6.0** > **Configure Tomcat**.
2. Select the **Java** tab. See screenshot below.
3. Add this line in the **Java Options** field: `-XX:MaxPermSize=256M`.
4. In the **Maximum memory pool field**, enter the value `1024`.

> The **Maximum memory pool** setting is equivalent to passing the `-Xmx` argument to Java.

To enable printing from Tomcat

When running Tomcat as a Windows Service, make sure that the user running the Tomcat service can use the printer. This is not the case by default. Permissions must be given to the tomcat user in order for Onyx to see the printer you wish to use.

> Can someone provide the way to properly do this? CaG fixed the issue by making Tomcat run as a domain user, then login to the server as that user and install the printer. There's probably a better way of doing this e.g.: install the printer as a "system" printer instead of installing it per-user

Configuring MySQL

To check the database connection settings

The default Onyx connection to the MySQL server uses the settings shown in the table below. They are defined in the `onyx-config.properti es` file, which is Onyx's main configuration file.

Setting	Value

hostname:port	localhost:3306
database name	onyx
username	onyx
password	onyx-demo

Your organization may have already changed the defaults. You can check the values in the `onyx-config.properties` file (see What You Need from Your Organization). If the settings are different from the defaults shown in the table, note them so you can use them when Creating a Database for Onyx in MySQL.

WEB-INF/config/onyx-config.properties

```
# Database configuration (if applicable)
[...]
# MySQL
org.obiba.onyx.datasource.driver=com.mysql.jdbc.Driver
org.obiba.onyx.datasource.url=jdbc:mysql://localhost:3306/onyx
org.obiba.onyx.datasource.username=onyx
org.obiba.onyx.datasource.password=onyx-demo
org.obiba.onyx.datasource.dialect=org.hibernate.dialect.MySQL5InnoDBDialect
org.obiba.onyx.datasource.validationQuery=SELECT 1;
org.obiba.onyx.datasource.testOnBorrow=true
```

To create a database for Onyx in MySQL

You must create a database for Onyx in MySQL, and you must also set up a user that has all privileges on that database. When Onyx starts for the first time, it will automatically create its schema in this database.

In a MySQL client:

1. Execute the command: `create database onyx` (or whatever name that defined at the end of this line in onyx-config.properties: `org.obiba.onyx.datasource.url=jdbc:mysql://localhost:3306/`**onyx**})
2. Ensure that the MySQL user (`onyx` in this case) is granted all privileges on that database instance (CREATE TABLE, ALTER, and so on).

Alternatively, in the MySQL Administrator application:

1. Select **Catalogs**.
2. Right-click in the field that lists the existing schemata, and select **Create New Schema**.
3. When prompted for a name for the new schema, enter `onyx` or whatever name was defined at the end of this line in onyx-config.properties: `org.obiba.onyx.datasource.url=jdbc:mysql://localhost:3306/`**onyx**).
4. Add a new user (with the name `onyx` or whatever value is defined in this line of onyx-config.properties: `org.obiba.onyx.datasource.username=onyx`).
5. Assign all priveleges on the new database to the new user.

Setting Up an SSL Connection

To set up Onyx to run over a secured connection on the local network, you must do two tasks on the Onyx server: create a keystore and configure Tomcat to use an SSL connection.

The procedures in this section use the following shortcuts to refer to certain directories:

- JAVA_HOME = the directory where java is installed
- CATALINA_BASE = the directory where Tomcat is installed

To create a keystore for the SSL connection

1. From the JAVA_HOME directory, execute this command:

```
./bin/keytool -genkey -alias tomcat -keyalg RSA -keystore
CATALINA_BASE/keystore/onyx.jks
```

2. When prompted, enter the requested information for the certificate and a new password.
3. Export the certificate to a file. You will import this file on each of the client workstations. See Setting Up the SSL Connection to Onyx

To configure Tomcat to use an SSL connection

Edit the file `CATALINA_BASE\...\conf\server.xml` as follows:

1. Comment out this section:

```
<!-- Define a non-SSL HTTP/1.1 Connector on port 8080 -->
    <!-- <Connector port="8080" maxHttpHeaderSize="8192"
               maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
               enableLookups="false" redirectPort="8443" acceptCount="100"
               connectionTimeout="20000" disableUploadTimeout="true" /> -->
```

2. Remove comments from this section:

```
<!-- Define a SSL HTTP/1.1 Connector on port 8443 -->
    <Connector port="8443" maxHttpHeaderSize="8192"
               maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
               enableLookups="false" disableUploadTimeout="true"
               acceptCount="100" scheme="https" secure="true"
               clientAuth="false" sslProtocol="TLS"/>
```

3. In the `<Connector>` element, add the following attributes:

```
keystoreFile="keystore\onyx.jks" keystorePass="password"
```

where `"keystore\onyx.jks"` and `"password"` are the values you entered when Creating a keystore for the SSL connection.

Generating a Key and Certificate for Data Export

Onyx can export data to one or more export destinations (see Configuring Data Export and Purge). If your organization has decided to encrypt participant data upon export to a particular destination, you must generate a key and certificate for that destination.

You can check whether or not data is supposed to be encrypted on export, look in the file `export-destinations.xml` which is in your *custom -onyx.war* file (What You Need from Your Organization). If any of the destinations include an `<encrypt>` element, the data will be encrypted on export.

You will need the key and certificate when you carry out the procedure for Deploying Onyx. You can use a utility (like openSSL) to generate keys and certificates. For a Windows program that uses openSSL to generate keys and certificates, see Shining Light Productions.

Fine-tuning the Onyx Configuration

The global configuration settings for Onyx are contained in the file `onyx-config.properties`. See What You Need from Your Organization.

As a minimum, you should check the configuration settings listed in this section. If you think other settings need to be fine-tuned, see Customizing the Global Configuration in the Onyx Customization & Configuration Guide.

To check the application mode

Onyx should be configured to run in deployment mode. This is very important since it improve the overall performance of the application. Check that the following property `onyx-config.properties` is set to `deployment` in :

WEB-INF/config/onyx-config.properties

```
# Onyx Web application mode: deployment or development
org.obiba.onyx.webapp.configurationType=deployment
```

To set the printer

On startup, Onyx decides which printer to use. Onyx will try to find a printer with a particular name. If that printer does not exist, or does not support PostScript printing, then Onyx will fall back to using the system's default printer. If that printer does not support PostScript printing, Onyx will not be able to print reports.

You can change the printer that Onyx should look for at startup by editing this line in `onyx-config.properties`.

WEB-INF/config/onyx-config.properties

```
# Name of printer for PDF printing.
org.obiba.onyx.pdfPrinterName=ONYX PDF Printer
```

To set up the appointment list

1. Check the input and output directories for the appointment list file. The output directory is optional. The default paths are shown below. They may have been changed in your version of Onyx.

WEB-INF/config/onyx-config.properties

```
# Appointment management
# Directory that contains the appointment list files to process
dropped by external process
org.obiba.onyx.appointments.inputDirectory=WEB-INF/appointments/in
# Optional directory that contains the successfully processed files
org.obiba.onyx.appointments.outputDirectory=WEB-INF/appointments/out
# Schedule for automatic appointment list updates (4am every day)
org.obiba.onyx.appointments.schedule=0 0 4 * * ?
```

---

2. Create the input and output directories on the Onyx server.
3. If your organization gave you an appointment list file, put it in the input directory.

To set up the data export directory

1. Check the export directory setting in `onyx-config.properties`. The default export directory is `target` in Onyx's current working directory (usually the Tomcat home directory). It may been changed in your version of Onyx.

WEB-INF/config/onyx-config.properties

```
# System path where to export Onyx Data
org.obiba.onyx.export.path=target
```

---

2. Create the export directory on the Onyx server.

To set up the keystore for data export

1. Check where the keystore file should be written to. The default path and filename are shown below. They may have been changed in your version of Onyx.

> **WEB-INF/config/onyx-config.properties**
> ```
> # Keystore
> org.obiba.onyx.keystore.file=file:${java.io.tmpdir}/onyxKeyStore.jks
> org.obiba.onyx.keystore.password=youshouldchangethispassword
> ```

2. Change the password for the keystore and save `onyx-config.properties`.
3. Create the keystore directory on the Onyx server.

Deploying Onyx

To deploy Onyx, you need a war file (*custom-onyx*.war) containing a customized version of Onyx. If you do not already have it, see here

> You must enter the *custom-onyx* part of the war filename when you access Onyx in a browser in the procedure below.

Copy the war file to the `webapps` directory in the Tomcat installation directory `$TOMCAT_HOME/webapps`.

1. If Tomcat is already running, it should deploy Onyx automatically. To check if Tomcat is running on Windows:
    1. Select **Control Panel > Administrative Tools > Services > Apache Tomcat**).
    2. Click **Start** if Tomcat is not running. Onyx will be deployed when Tomcat starts.
2. Check that you can access Onyx by opening a browser and pointing it to: http://localhost:8080/*custom-onyx*. If you cannot access Onyx, try restarting Tomcat. If you still have trouble, see Troubleshooting.
3. When you access Onyx for the first time, you must complete the Onyx setup page (see the screenshot below) as follows:
    1. Enter details about the Onyx instance (study name, site name and site id).
    2. If you configured Onyx to Generate Participant Identifiers Automatically, you will have to specify the Identifier Prefix and the First Indentifier (starting point for generating the identifiers). See Configuring Participant ID Generator for more details about generating identifiers.
    3. Set up an account for the Onyx Administrator.
    4. Set the Session Timeout (minutes) to at least 45 or 60 (which should be enough for any physical measurements that must be performed).
    5. If your organization will encrypt participant data upon export, copy and paste the certificate you generated for this purpose into the field provided. If you did not generate a certificate yet, see Generating a Key and Certificate for Data Export.

> Take note of these values for future reference:
>
> - Administrator's user name and password. You will always need them to log in to Onyx.
> - Collection Site Id. This value will be used in the appointment list file. If your organization has or will have several sites (each with its own Onyx server), each site should have a unique Id. The value can be alphanumeric, but should not contain spaces.

Checking the Deployment

Here are a few items to check after you have deployed Onyx:

- If your data will be encrypted upon export (see Generating a Key and Certificate for Data Export), check that the file `onyxKeyStore.jks` was created at the location defined by the property `org.obiba.onyx.keystore.file` in `onyx-config.properties`.
- If you put an appointment list file in the input directory (see Setting up the appointment list), check that you can update the appointment

list from the **Participants** tab of the Onyx user interface.

Troubleshooting

The first place to look is Tomcat's console output which is written in the `log` directory. Under windows, the file is called `stdout_XXX.log` on Debian/Ubuntu, this file is called `catalina.out`.

Miscellaneous

Consent Form Submission Problems

You may encounter an error when attempting to submit a consent form, especially a large multi-page consent form written in a double-byte character set. This error occurs because the consent form is too large to be stored in the database. To get around this, you must modify the MySql database settings to allow storage of larger amounts of data. The configuration file that you must edit is `my.ini` on Windows and `my.cnf` on Linux. This is the line you need to change:

```
max_allowed_packet=3M
```

To start, try a value of `3M` (for 3 megabytes). If it still doesn't work, increase the value. You will need to restart the MySql database server after making the change.

Setting Up the Workstations

Overview

Setting up the Onyx client workstations involves some or all of the following tasks on each workstation:

- Setting Up Touchscreens (if applicable)
- Registering Instruments
- Setting Up Electronic Instruments (if applicable)
    - Installing Proprietary Software for Instruments
    - General Troubleshooting for Instruments
        - BadFieldException when Starting an Instrument
        - Unable to Save Instrument Data Due to Timeout of Onyx Session
    - Tanita Scales and Bioimpedance Instruments
        - Cable for the Tanita scale
    - Achilles Express
    - TOPCON TRC-nw8

Setting Up Touchscreens (if applicable)

If touchscreen monitors are being used with any of the client workstations, the monitors should be put in "Click on contact" mode. You should avoid any kind of "mouse emulation" mode, since it will complicate use of the touchscreen.

Registering Instruments

An instrument must be registered on the workstation(s) with which it will be used. An instrument can be reserved for one workstation or it can be shared by several workstations. This is done on the **Workstation** page of the Onyx user interface, as explained in the Onyx User Guide.

Setting Up Electronic Instruments (if applicable)

Onyx supports physical measurements taken manually and electronically. Onyx supports certain electronic instruments (it launches the instrument's software and captures measurements from the instrument). If your organization has chosen to use any of these electronic instruments, check this section for any setup information that applies to them.

Installing Proprietary Software for Instruments

If your organization has chosen to use any of these electronic instruments (except those manufactured by Tanita), you must install the instrument's proprietary software on the workstation with which the instrument will be used.

General Troubleshooting for Instruments

BadFieldException **when Starting an Instrument**

This exception is caused by a bug in the version of Java installed on the workstation. To fix this issue, ensure that the JRE installed on the Onyx server is at least `1.6.0_02`.

Unable to Save Instrument Data Due to Timeout of Onyx Session

After an instrument's software is launched, the user's Onyx session may timeout because the user is no longer interacting with Onyx. If the Onyx timeout is set too low (for example, `5` minutes) and the physical measurement takes longer than the timeout, the Onyx session will timeout and the measurement data will not be saved. The user will need to log in to Onyx again after the test is over, and repeat the measurement or enter the measurements in Onyx manually, if Onyx is configured to accept manual measurments.

The timeout is set when you access Onyx for the first time (see Installing Onyx on the Server). The timeout setting is stored in the Onyx database. It cannot be changed in a configuration file or in the Onyx user interface.

When accessing Onyx for the first time, be sure to set the Session Timeout to at least 45 or 60 (minutes). This should be enough to complete physical measurements. See Setting Up the Server.

Tanita Scales and Bioimpedance Instruments

If your organization is using instruments manufactured by Tanita, read this section when you are setting up the instrument(s) for use with an Onyx workstation.

Cable for the Tanita scale

If the cable provided has a serial 9 pin connector, the Port Settings for Flow control must be set to Hardware.

Configuring the Port Settings on Windows

1. Open the **Control Panel** in classic view.
2. Select **System > Hardware (tab) > Device Manager > Expand Ports**.
3. Double-click the port that needs to be changed (usually COM1).
4. Select the **Port Settings** tab.
5. Set **Flow Control** to **Hardware**.

Achilles Express

Achilles Express software (version 5.11 and earlier versions) produces erroneous data. Specifically, the SOS and BUA parameters are not calculated properly. However, the Stiffness Index is correct.

Version 5.12 fixes this issue.

TOPCON TRC-nw8

If SQL Server uses windows authentication to connect to the database, "ntmlauth.dll" library must be added in lib java folder.
This library is provided by jtds driver located at `http://jtds.sourceforge.net/`.
Once zip archive downloaded (jtds-1.2.x-dist.zip), DLL library is in {x86, x64, IA64}/SSO/ntmlauth.dll (Platform Dependant).

Onyx User Guide

View the latest version of the *Onyx User Guide* in PDF format.

Contents of this Guide

Introduction

If you are new to Onyx and would like a gentle introduction, this chapter is for you.
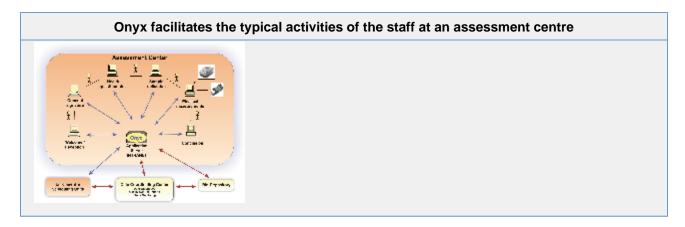
If you are already familiar with Onyx and need specific information about how to do something, skip this chapter and find the topic you need by scanning the table of contents or by searching for some key words that describe what you are looking for.

This chapter covers these topics:

- What Onyx is
- Key concepts related to the way Onyx works
- What Onyx does
- Who makes Onyx
- Onyx documentation
- Onyx user support

What is Onyx?

Onyx is a web-based application used to manage participant baseline interviews by assessment centres and clinics that are collecting data for research. Typically, the data is being collected for biobanks or large-scale population studies. this figure shows the typical assessment centre activities that Onyx supports. As shown in the figure, Onyx is installed on a server in the assessment centre and can be accessed from workstations via the assessment centre's intranet. An assessment centre may have one or more workstations dedicated to collecting data using Onyx.

Onyx facilitates the typical activities of the staff at an assessment centre

A Highly Customizable Software Solution

Onyx is modular as shown in this figure. The Onyx engine is a backbone into which independent data collection components are inserted. Onyx is *configurable* which means certain stages may or may not be included in your version of Onyx. For example, your study may not accept volunteer participants. And if a stage is included, it can be fine-tuned to meet the requirements of your study. For example, for your study Onyx may be configured to only accept electronic signatures for consent forms, whereas for another study Onyx may accept electronic or handwritten signatures. Onyx is also *customizable* which means certain stages are tailor-made for each research study. For example, each study develops its own questionnaires, and each study defines which physical measurements will be collected and in which order.

Key Concepts

Participant

A participant is a person who has come to the assessment centre in order to participate in the research study. Assessment centre staff collect data from participants by asking questions, collecting biospecimens, and taking physical measurements.

Each research study determines how they will recruit participants. Typically, participants are either invited or volunteers. Onyx can be customized to accept volunteer participants or not. Whether or not this functionality appears in Onyx at your assessment centre depends on how Onyx was configured for your study. See Receiving vs. Enrolling Participants.

Invited Participant

A participant who was selected by the study and given an appointment time for their interview.

Volunteer Participant

A volunteer participant (also known as a "walk-in" Walk-in participants) arrives at the assessment centre without an appointment. They heard about the study in some way and decided that they would like to participate.

Interview

Onyx treats an interview as a set of interdependent stages that a participant goes through in order to complete their visit to the assessment centre. See this figure. An interview starts when a staff member receives the participant and enters registration information for the participant in Onyx. The participant then passes through some or all the stages that have been defined for the study. The interview ends when a staff member clicks the `Close` interview button in Onyx.

Onyx treats an interview as a set of interdependent stages

Interview Stages

An interview stage is one of the interdependent parts of an interview during which a particular type of data is collected from the participant. Each research study defines the stages it requires to obtain the necessary data, as well as the sequence in which the stages appear in Onyx. Typical interview stages include:

- Signing a consent form
- Questionnaires (one or more)
- Physical measurements (one or more)
- Collection of biospecimens (one or more)
- Conclusion of the interview (may include printing a report for the participant that includes information considered appropriate by the research study)

Each research study defines its own stages, creates its own questionnaires, and decides which physical measurements and biospecimens must be collected. Onyx stores the data collected during the stages centrally and makes it available to all workstations.

Who Uses Onyx

Onyx is used by the staff of an assessment center or a clinic that is collecting data from participants in a biobank study.

Onyx supports four roles for users. An individual user can be assigned one or several of these roles:

- Data Collectors
- Participant Managers
- Receptionists
- Questionnaire Editors
- System Administrators

Data Collectors

These Onyx users are responsible for one or more of these tasks:

- interviewing participants
- collecting biospecimens
- taking physical measurements

Onyx data collectors are typically nurses and technologists.

Participant Managers

These Onyx users coordinate the activities of data collectors. Participant managers can do the tasks of a data collector, and in addition, they can do these tasks :

- receiving participants
- enrolling volunteer participants (if your study permits this)
- updating the appointment list
- unlocking a locked interview

Receptionists

These Onyx users can do the following tasks:

- receiving participants
- enrolling volunteer participants (if your study permits this)
- updating the appointment list

Questionnaire Editors

These Onyx users create and edit the questionnaires that form part of an Onyx interview. Typically, they would be researchers involved in the design of the study, but not in interviewing participants. This role only gives a user the permission to create and edit questionnaires.

System Administrators

These Onyx users are typically Information Technology Managers (IT Managers) who handle the technical tasks required to set up and maintain Onyx. This role gives a user the permission to manage users and data.

What Onyx Does

Onyx facilitates the work of the staff members by allowing them to collect data electronically:

- obtain participant consent with the help of an electronic signature pad (if your assessment centre does not have a signature pad or the pad is not available, you record the fact that the patient signed a printed consent form in Onyx)
- complete onscreen questionnaires with participants
- set up participants to answer self-administered questionnaires on a touchscreen or a workstation
- record the collection of biospecimens (using a barcode scanner to identify the sample or entering a sample number manually)
- register physical measurementssuch as height, weight, bone density, and any other measurements that the study may require

Onyx provides researchers with tools that allow them to optimize the workflow of baseline interviews:

- control stage availability and dependencies between stages (some examples: require that consent was obtained before an interview can proceed; prevent staff from taking measurements or samples that are contraindicated; ensure that dependent measurements are done in the correct order)
- capture administrative parameters such as start and end times of each stage
- automate the calibration of electronic instruments used to take physical measurements
- produce personalized reports for participants
- export encrypted data to multiple destinations

Some things Onyx does not do

While Onyx includes many features and functions that an assessment centre needs, it does not include the following functionality:

- Onyx does not allow you to schedule participant appointments. The appointment list must be imported into Onyx.
- Onyx does not arbitrarily decide when and whether you can proceed with a certain stage of an interview. Onyx is highly configurable and customizable. Each research study defines its own questionnaires, the physical measurments and biospecimens to be collected, the order of interview stages, and the conditions for passing from one stage to another. Based on how your research study configured Onyx, Onyx may inform you that a stage is contraindicated or prompt you about what to do during a certain stage. For example because of the participant's answers to certain questions, Onyx might prompt you about how to measure the participan'ts blood pressure: `Use the participant's right arm to take blood pressure`
- Onyx does not allow you to analyze the data collected from participants. It can export the data to other destinations where the analysis can be done.
- Onyx is not a Laboratory Information Management System (LIMS). It does not track processing of biospecimens.

Who Makes Onyx

Onyx is developed by OBiBa, a collaborative international project whose mission is to build high-quality open source software for biobanks. To learn more about OBiBa, please have a look at our website.

Onyx Documentation

In addition to this PDF version of the *Onyx User Guide*, an Online Help version of this guide is being prepared.

Information about configuring Onyx can be found in the Onyx Configuration Guide.

Version of Onyx Covered by this Guide

This version of the *Onyx User Guide* describes Onyx 1.8.0, in particular, version 1.8.0 of Onyx.

How this Guide is Organized

Since most Onyx users are data collectors, most of this guide explains how to use Onyx to collect participant data. This guide also includes chapters for Participant Managers, Questionnaire Editors, and for System Administrators.

This guide consists of the following chapters:

- **Chapter 1: Introduction.** Presents Onyx. Includes key concepts that will help you understand Onyx's approach to baseline interviews. *For new Onyx users.*
- **Chapter 2: Getting Started.** Presents the Onyx `Home` page. Explains how to log in to Onyx and the simplest way to start an interview. *For new Onyx users.*
- **Chapter 3: Viewing Participants.** Presents the `Participants` page and key concepts related to viewing participants. Explains the ways you can search for a participant in Onyx. *For new Onyx users.*
- **Chapter 4: Managing an Interview.** Presents the `Interview` page and key concepts related to navigating through an interview. Covers various ways to access and exit the stages of an interview: starting, stopping, pausing, resuming, and so on. *Of most interest to Onyx data collectors and participant managers.*
- **Chapter 5: Obtaining Participant Consent.** Presents Onyx's way of handling participant consent. Covers electronic consent forms and registering paper consent. *Of most interest to Onyx data collectors and participant managers.*
- **Chapter 6: Completing Questionnaires.** Presents key concepts related to the way questionnaires are handled in Onyxsuch as assisted versus self-administered questionnaires. Since each study designs its custom questionnaires, the chapter explains the types of questions and answers users will see, rather than how to answer the particular questions in your study's questionnaires. *Of most interest to Onyx data collectors and participant managers.*
- **Chapter 7: Collecting Physical Measurements.** Presents key concepts related to physical measurement stages. Since each study determines the physical measurements it requires, the chapter explains the general workflow of a physical measurement stage and demonstrates the workflow with an example stage. *Of most interest to Onyx data collectors and participant managers.*
- **Chapter 8: Collecting Biospecimens.** Presents key concepts related to stages used to record the collection of biospecimens. Since each study determines the biospecimens it requires, the chapter provides an example biospecimen collection stage. *Of most interest to Onyx data collectors and participant managers.*
- **Chapter 9: Managing Your Onyx User Profile.** Explains how to do a few tasks that customize Onyx for you: changing your password and changing the language of the Onyx user interface (English or French are currently available). *For all Onyx users.*
- **Chapter 10: Managing an Onyx Workstation.** Presents the Workstation page and key concepts related to storing data about a workstation. Explains how to register and calibrate instruments used for physical measurements, and how to maintain logs of experimental conditions. *Of most interest to Onyx users who will register and calibrate instruments, and log experimental conditions.*
- **Chapter 11: Creating and Editing Questionnaires.** Explains how to create and edit questionnaires that can be included as stages of an Onyx interview. This work is typically done by researchers involved in design of the study. *Of interest to users with the role of questionnaire editor.*
- **Chapter 12: Topics for Participant Managers.** Explains certain tasks that only participant managers can dosuch as receiving and enrolling participants. *Of most interest to participant managers.*
- **Chapter 13: Topics for System Administrators.** Explains certain tasks that only system administrators can dosuch as managing Onyx users, and exporting and purging data. *Of most interest to system administrators.*

Icons Used in this Guide

Key Concept: This icon appears beside explanations of key concepts. In most chapters of this guide, you will find explanations of key concepts relevant to the chapter.

Customizable: This icon appears besides notes that explain a feature of Onyx that can be customized. Since each study customizes Onyx for its needs, these sections explain why you may not see a certain feature in your version of Onyx.

Procedure: This icon appears beside proceduresstep-by-step instructions for performing a task in Onyx.

Pointer: This icon appears beside suggestions that could make your work in Onyx a little easier.

Onyx Support

- You can reach Onyx customer support by email at: support@obiba.org
- You can join the OBiBa users group. New releases of Onyx are announced through this group. You can use this forum to make comments, to ask questions, and to share ideas with other users of OBiBa software.
- You can visit the OBiBa website where you will find:
    - News stories and presentations about Onyx and other OBiBa products
    - JIRA, an issue-tracking system that allows you to enter bug reports, request new features, and suggest ways to improve Onyx

Introduction

Introduction

If you are new to Onyx and would like a gentle introduction, this chapter is for you.
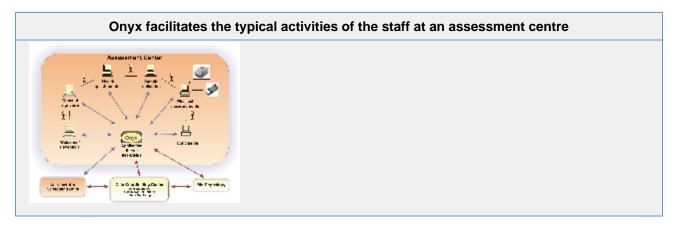
If you are already familiar with Onyx and need specific information about how to do something, skip this chapter and find the topic you need by scanning the table of contents or by searching for some key words that describe what you are looking for.

This chapter covers these topics:

- What Onyx is
- Key concepts related to the way Onyx works
- What Onyx does
- Who makes Onyx
- Onyx documentation
- Onyx user support

What is Onyx?

Onyx is a web-based application used to manage participant baseline interviews by assessment centres and clinics that are collecting data for research. Typically, the data is being collected for biobanks or large-scale population studies. this figure shows the typical assessment centre activities that Onyx supports. As shown in the figure, Onyx is installed on a server in the assessment centre and can be accessed from workstations via the assessment centre's intranet. An assessment centre may have one or more workstations dedicated to collecting data using Onyx.

Onyx facilitates the typical activities of the staff at an assessment centre

A Highly Customizable Software Solution

Onyx is modular as shown in this figure. The Onyx engine is a backbone into which independent data collection components are inserted. Onyx is *configurable* which means certain stages may or may not be included in your version of Onyx. For example, your study may not accept volunteer participants. And if a stage is included, it can be fine-tuned to meet the requirements of your study. For example, for your study Onyx may be configured to only accept electronic signatures for consent forms, whereas for another study Onyx may accept electronic or handwritten signatures. Onyx is also *customizable* which means certain stages are tailor-made for each research study. For example, each study develops its own questionnaires, and each study defines which physical measurements will be collected and in which order.

Key Concepts

Participant

A participant is a person who has come to the assessment centre in order to participate in the research study. Assessment centre staff collect data from participants by asking questions, collecting biospecimens, and taking physical measurements.

Each research study determines how they will recruit participants. Typically, participants are either invited or volunteers. Onyx can be customized to accept volunteer participants or not. Whether or not this functionality appears in Onyx at your assessment centre depends on how Onyx was configured for your study. See Receiving vs. Enrolling Participants.

Invited Participant

A participant who was selected by the study and given an appointment time for their interview.

Volunteer Participant

A volunteer participant (also known as a "walk-in" Walk-in participants) arrives at the assessment centre without an appointment. They heard about the study in some way and decided that they would like to participate.

Interview

Onyx treats an interview as a set of interdependent stages that a participant goes through in order to complete their visit to the assessment centre. See this figure. An interview starts when a staff member receives the participant and enters registration information for the participant in Onyx. The participant then passes through some or all the stages that have been defined for the study. The interview ends when a staff member clicks the `Close` interview button in Onyx.

Onyx treats an interview as a set of interdependent stages

Interview Stages

An interview stage is one of the interdependent parts of an interview during which a particular type of data is collected from the participant. Each research study defines the stages it requires to obtain the necessary data, as well as the sequence in which the stages appear in Onyx. Typical interview stages include:

- Signing a consent form
- Questionnaires (one or more)
- Physical measurements (one or more)
- Collection of biospecimens (one or more)
- Conclusion of the interview (may include printing a report for the participant that includes information considered appropriate by the research study)

Each research study defines its own stages, creates its own questionnaires, and decides which physical measurements and biospecimens must be collected. Onyx stores the data collected during the stages centrally and makes it available to all workstations.

Who Uses Onyx

Onyx is used by the staff of an assessment center or a clinic that is collecting data from participants in a biobank study.

Onyx supports four roles for users. An individual user can be assigned one or several of these roles:

- Data Collectors
- Participant Managers
- Questionnaire Editors
- System Administrators

Data Collectors

These Onyx users are responsible for one or more of these tasks:

- interviewing participants
- collecting biospecimens
- taking physical measurements

Onyx data collectors are typically nurses and technologists.

Participant Managers

These Onyx users coordinate the activities of data collectors. Participant managers can do the tasks of a data collector, and in addition, they can do these tasks :

- receiving participants
- enrolling volunteer participants (if your study permits this)
- updating the appointment list
- unlocking a locked interview

Questionnaire Editors

These Onyx users create and edit the questionnaires that form part of an Onyx interview. Typically, they would be researchers involved in the

design of the study, but not in interviewing participants. This role only gives a user the permission to create and edit questionnaires.

System Administrators

These Onyx users are typically Information Technology Managers (IT Managers) who handle the technical tasks required to set up and maintain Onyx. This role gives a user the permission to manage users and data.

What Onyx Does

Onyx facilitates the work of the staff members by allowing them to collect data electronically:

- obtain participant consent with the help of an electronic signature pad (if your assessment centre does not have a signature pad or the pad is not available, you record the fact that the patient signed a printed consent form in Onyx)
- complete onscreen questionnaires with participants
- set up participants to answer self-administered questionnaires on a touchscreen or a workstation
- record the collection of biospecimens (using a barcode scanner to identify the sample or entering a sample number manually)
- register physical measurementssuch as height, weight, bone density, and any other measurements that the study may require

Onyx provides researchers with tools that allow them to optimize the workflow of baseline interviews:

- control stage availability and dependencies between stages (some examples: require that consent was obtained before an interview can proceed; prevent staff from taking measurements or samples that are contraindicated; ensure that dependent measurements are done in the correct order)
- capture administrative parameters such as start and end times of each stage
- automate the calibration of electronic instruments used to take physical measurements
- produce personalized reports for participants
- export encrypted data to multiple destinations

Some things Onyx does not do

While Onyx includes many features and functions that an assessment centre needs, it does not include the following functionality:

- Onyx does not allow you to schedule participant appointments. The appointment list must be imported into Onyx.
- Onyx does not arbitrarily decide when and whether you can proceed with a certain stage of an interview. Onyx is highly configurable and customizable. Each research study defines its own questionnaires, the physical measurments and biospecimens to be collected, the order of interview stages, and the conditions for passing from one stage to another. Based on how your research study configured Onyx, Onyx may inform you that a stage is contraindicated or prompt you about what to do during a certain stage. For example because of the participant's answers to certain questions, Onyx might prompt you about how to measure the participan'ts blood pressure: `Use the participant's right arm to take blood pressure`
- Onyx does not allow you to analyze the data collected from participants. It can export the data to other destinations where the analysis can be done.
- Onyx is not a Laboratory Information Management System (LIMS). It does not track processing of biospecimens.

Who Makes Onyx

Onyx is developed by OBiBa, a collaborative international project whose mission is to build high-quality open source software for biobanks. To learn more about OBiBa, please have a look at our website.

Onyx Documentation

In addition to this PDF version of the *Onyx User Guide*, an Online Help version of this guide is being prepared.

Information about configuring Onyx can be found in the Onyx Configuration Guide.

Version of Onyx Covered by this Guide

This version of the *Onyx User Guide* describes Onyx 1.8.0, in particular, version 1.8.0 of Onyx.

How this Guide is Organized

Since most Onyx users are data collectors, most of this guide explains how to use Onyx to collect participant data. This guide also includes chapters for Participant Managers, Questionnaire Editors, and for System Administrators.

This guide consists of the following chapters:

- **Chapter 1: Introduction.** Presents Onyx. Includes key concepts that will help you understand Onyx's approach to baseline interviews. *F or new Onyx users.*
- **Chapter 2: Getting Started.** Presents the Onyx `Home` page. Explains how to log in to Onyx and the simplest way to start an interview. *Fo r new Onyx users.*

- **Chapter 3: Viewing Participants.** Presents the `Participants` page and key concepts related to viewing participants. Explains the ways you can search for a participant in Onyx. *For new Onyx users.*
- **Chapter 4: Managing an Interview.** Presents the `Interview` page and key concepts related to navigating through an interview. Covers various ways to access and exit the stages of an interview: starting, stopping, pausing, resuming, and so on. *Of most interest to Onyx data collectors and participant managers.*
- **Chapter 5: Obtaining Participant Consent.** Presents Onyx's way of handling participant consent. Covers electronic consent forms and registering paper consent. *Of most interest to Onyx data collectors and participant managers.*
- **Chapter 6: Completing Questionnaires.** Presents key concepts related to the way questionnaires are handled in Onyxsuch as assisted versus self-administered questionnaires. Since each study designs its custom questionnaires, the chapter explains the types of questions and answers users will see, rather than how to answer the particular questions in your study's questionnaires. *Of most interest to Onyx data collectors and participant managers.*
- **Chapter 7: Collecting Physical Measurements.** Presents key concepts related to physical measurement stages. Since each study determines the physical measurements it requires, the chapter explains the general workflow of a physical measurement stage and demonstrates the workflow with an example stage. *Of most interest to Onyx data collectors and participant managers.*
- **Chapter 8: Collecting Biospecimens.** Presents key concepts related to stages used to record the collection of biospecimens. Since each study determines the biospecimens it requires, the chapter provides an example biospecimen collection stage. *Of most interest to Onyx data collectors and participant managers.*
- **Chapter 9: Managing Your Onyx User Profile.** Explains how to do a few tasks that customize Onyx for you: changing your password and changing the language of the Onyx user interface (English or French are currently available). *For all Onyx users.*
- **Chapter 10: Managing an Onyx Workstation.** Presents the Workstation page and key concepts related to storing data about a workstation. Explains how to register and calibrate instruments used for physical measurements, and how to maintain logs of experimental conditions. *Of most interest to Onyx users who will register and calibrate instruments, and log experimental conditions.*
- **Chapter 11: Creating and Editing Questionnaires.** Explains how to create and edit questionnaires that can be included as stages of an Onyx interview. This work is typically done by researchers involved in design of the study. *Of interest to users with the role of questionnaire editor.*
- **Chapter 12: Topics for Participant Managers.** Explains certain tasks that only participant managers can dosuch as receiving and enrolling participants. *Of most interest to participant managers.*
- **Chapter 13: Topics for System Administrators.** Explains certain tasks that only system administrators can dosuch as managing Onyx users, and exporting and purging data. *Of most interest to system administrators.*

Icons Used in this Guide

Key Concept: This icon appears beside explanations of key concepts. In most chapters of this guide, you will find explanations of key concepts relevant to the chapter.

Customizable: This icon appears besides notes that explain a feature of Onyx that can be customized. Since each study customizes Onyx for its needs, these sections explain why you may not see a certain feature in your version of Onyx.

Procedure: This icon appears beside proceduresstep-by-step instructions for performing a task in Onyx.

Pointer: This icon appears beside suggestions that could make your work in Onyx a little easier.

Onyx Support

- You can reach Onyx customer support by email at: support@obiba.org
- You can join the OBiBa users group. New releases of Onyx are announced through this group. You can use this forum to make comments, to ask questions, and to share ideas with other users of OBiBa software.
- You can visit the OBiBa website where you will find:
    - News stories and presentations about Onyx and other OBiBa products
    - JIRA, an issue-tracking system that allows you to enter bug reports, request new features, and suggest ways to improve Onyx

Getting Started

Getting Started

This chapter is your starting point for jumping in and starting to use Onyx. Sit down in front of an Onyx workstation and try the things described in this chapter.

This chapter covers these topics:

- How to log in
- Onyx home page and the features it shares with other pages of Onyx

- How to start an interview

⚠ Logging In

The first step in using Onyx is *logging in* Logging inSigning inalso known as *signing in* Signing in. You must log in at the beginning of your work day or shift. When you have finished your work session, you should log out.
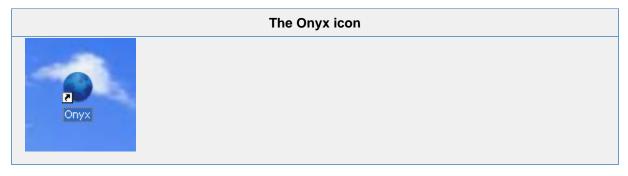
If you stay logged in for an extended period of time without touching the keyboard or mouse, Onyx automatically logs you out Logoutautomatic. Sometimes this is called "timing out" timing out. If Onyx logs you out, you must log in again as explained in this section.

Prerequisites
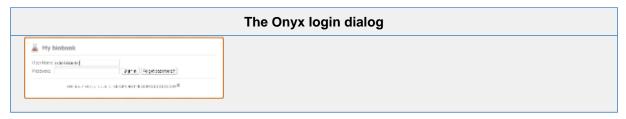
To log in, you need to know your user name and password. If you do not know them, speak to your manager or system administrator.

Procedure

1. If the Onyx login dialog (this figure) is not displayed on your workstation, locate the `Onyx` icon (this figure) on the desktop of your workstation.

The Onyx icon
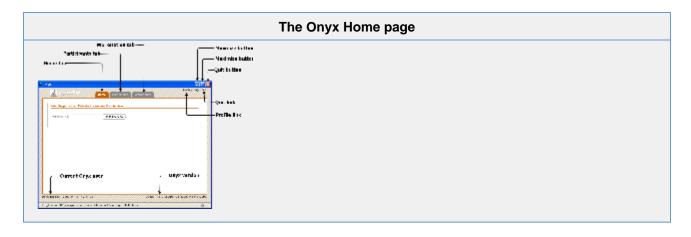
2. Double-click the `Onyx` icon. The Onyx login dialog should now be displayed on your workstation.

The Onyx login dialog

3. Enter your user name and password and click the `Sign in` button. The Onyx `Home` page is displayed. See this figure.

The Onyx `Home` Page

After you log in to Onyx, the `Home` page is displayed. This is the page that you will use to start or resume the interview of a participant who has already been received at the assessment centre. this figure shows the `Home` page and points out its main features. These features are described below this figure.

The Onyx Home page

Features Common to the Home Page and Other Pages

These features appear around the edge of the `Home` page, as well as two other pages of Onyxthe `Participants` page and the `Workstation` page:

- `Home` tab displays the Home page in which you can select a participant to interview (if you know their ID or can scan their barcode)
- `Participants` tab display the Participants page in which you can view the list of participants and to select a participant to interview
- `Workstation` tab displays the Workstation page in which you can register and calibrate instruments connected to the workstation, and to keep logs of room and environmental conditions

  > The `Workstation` page might not be visible if your study did not define any physical measurement stages or experimental condition logs.

- **Current Onyx user** - the user name of the Onyx user who is currently logged in, and the date and time when they logged in
- **Current version of Onyx** the version of Onyx that you are currently using
- `Profile` link allows the current user to change their profile (language and password)
- `Quit` link allows you to close Onyx. Always exit the interview you are working on before clicking the `Quit` link.
- `Help` link not currently functional
- **Standard Windows buttons**:
    - `Minimize` button - allows you to minimize Onyx (to a button at the bottom of the screen)
    - `Maximize/Reduce` button - allows you to go back and forth between full-screen and reduced window size
    - `Quit button` button - allows you to close Onyx (same as the Quit link)

- **Keyboard shortcuts**:
    - To increase the font size of text: Press **Ctrl** and **+** keys.
    - To decrease the font size of text: Press the **Ctrl** and **-** keys.
    - To jump forward between fields and/or buttons: Press the **Tab** key.
    - To jump backward between fields and/or buttons: Press the **Shift** and **Tab** keys.

The Simplest Way to Start an Interview

Usually, a participant manager receives participants and registers them. During registration, a participant ID is assigned to the participant. After a participant has been registered and has a participant ID, you can start their interview as explained in this section.

After you start an interview, you will proceed through the various stages of the interview with the participant: obtaining consent, answering questionnaires, and so on. During a stage or between the stages, you may need to pause the interview. If an interview has been paused, you can resume it as explained in this section.

Prerequisites

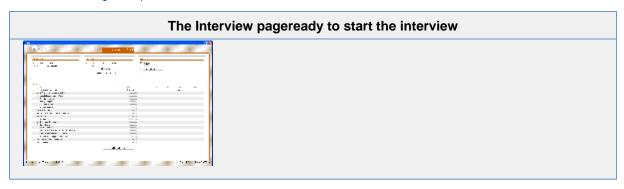To start or resume an interview for a participant:

- The participant must have been received or enrolled. Only participant managers can receive and enroll participants. See Receiving a participant and Enrolling a volunteer participant.
- You must know the participant's Participant ID or, if your assessment centre uses barcodes and you have access to a barcode scanner, you can scan the barcode used for the Participant ID. If you do not know the Participant ID, you can search for it, see Finding a participant.

Procedure

1. If the `Home` page is not displayed, click the `Home` tab.

The simplest way to start an interview is from the Home page

2. Enter the participant's Participant ID in the field on the `Home` page. Or, if your assessment centre uses barcodes, scan the participant's barcodethe number read by the scanner appears in the `Participant ID` field.Alternatively, if you know the participant's Enrollment ID, you may enter that instead.
3. Select the `Go to Interview` button. The `Interview` page is displayedsee this figure. You are now ready to proceed with the interviewsee Interviewing Participants.

The Interview pageready to start the interview

Viewing Participants

Viewing Participants
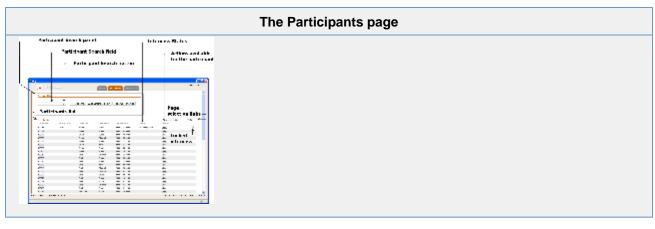
Onyx stores information about participants in its database on a server computer located in the assessment centre. All Onyx workstations can access the database. The `Participants` page allows you to view a list of the participants in the database and to find individual participants so that you can view or edit their registration information, and work on their interviews.

This chapter includes these topics related to viewing participants in the `Participants` page:

- The Participants page and its important features
- Key concepts related to viewing participants
- Procedures related to viewing participants
  - Displaying a list of all participants
  - Viewing appointments for the day
  - Viewing all interviews in progress
  - Finding a participant
  - Viewing a participant's registration information
  - Determining who has a lock on a participant

The Participants Page

When you select the `Participants` tab, the `Participants` page is displayed. This page gives you access to all the participants in the Onyx database. The `Home` page gives you one way of finding a participant. The `Participants` page lets you find a participant in a number of ways. this figure shows the `Participants` page and points out some of its more important features.

The Participants page

Here is an overview of the important features of the `Participants` page:

- The `Participants Search` panel — includes three buttons that give you easy access to participants that you often need to view:
    - the `All Participants` button - displays a list of all the participants in the Onyx database
    - the `Appointments of the day` button - displays a list of all the participants who have appointments on today's date
    - the `Interview in progress` button - displays a list of all the participants whose interviews have the status In progress
- The `Participant Search` field and 🔍 button - allow you to to search for a particular patient by name or ID
- The `Participants` list occupies the lower part of the page. Initially, it is empty, but after you do a search, it is filled with a list of participants.
- The `Actions` column contains one or more action links beside each participant. These are the actions you can do for the participant---for example: `View` their registration information and `Interview` them.
- The **Page selection links** (Showing 1 to 50 of 103 << < 1 2 3 > >>). If the `Participants` list contains many participants, they are displayed on several pages. These links allow you to view the different pages.

- A **Locked interview** (indicated by a small 🔒 icon) means that some user is interviewing the participant. See Locked status.

Key Concepts

This section contains key concepts related to viewing participants.

🖋 Participants List

A list of participants stored in the Onyx database.

When you are in the `Participants` page, you can choose to see some or all of the participants in the Onyx database by filtering the participants list in various ways: by name, enrollment ID, or participant ID. You can view all participants with appointments today or those currently being interviewed.

🖋 Appointment List

A list of appointments stored in a file or created in a programme other than Onyx.

Since Onyx does not allow you to schedule appointments for participants, it imports an appointment list from a file or another programme. Onyx users of the type participant manager can update the appointment list.

As a minimum, the appointment list contains the following information for each participant:

- participant name
- apointment date and time
- enrollment ID

The appointment list may contain other information depending on how Onyx was configured for your study.

🖋 Enrollment ID

A unique identifier for a participant that is imported into Onyx from the appointment list.

Enrollment IDs are either assigned automatically by the programme that schedules appointments, or they are defined in the file containing the appointment list. After the participant has been received, the Enrollment ID is not used very much in Onyx. Instead, the Participant ID is used to identify the participant during the interview. However, you can search for a participant by their Enrollment ID.

> You may notice that some participants do not have an Enrollment ID. This is probably because your study accepts Volunteer Participants. Volunteers do not have an enrollment ID since they were never on an appointment list.

Participant ID

A unique identifier for a participant that must be entered in Onyx when the participant arrives at the assessment centre and at various times during the interview.

Each study or assessment centre decides how it will assign Participant IDs and the format of its Participant IDs (the pattern of numbers and letters in the ID). Some studies use barcodes that can be scanned and displayed automatically in Onyx. Other studies will require you to enter the Participant ID manually.

Participant Barcode

A barcode that is used for the Participant ID.

Some studies use a barcode as the source of Participant IDs. The barcodes may be on a blood collection tube, an ID bracelet, or a label of some sort. If your study uses barcodes, a barcode scanner will be connected to your Onyx workstation so that you can easily obtain the Participant ID.

Interview Status

The current state of a participant's interview. See also Stage Status.

The status of a participant's interview appears in the `Status` column of the `Participants` list. Before a participant has been received, nothing appears in the status column for that participant. After a participant has been received, their interview can have one of these statuses:

- `In Progress` — This status means that the participant has been received and that their interview has been started. Even if the participant leaves the room for some reason (and a stage of their interview has been paused), the status of their interview remains `In Progress`. See Displaying a list of In Progress interviews.
- **Locked** — The lock icon 🔒 means another staff member is interviewing the participant. If you suspect that there is a problem or if you need to work on the interview, contact your participant manager who can remove the lock, if necessary. See Unlocking an Interview.
- `Completed` — This status means that the interview was completed and that all stages were completed successfully and the interview was concluded correctly by a staff member.
- `Closed` — This status means that the interview was closed without all the stages being completed. An interview can be closed at any stage, if necessary. A closed interview cannot be restarted. The data collected cannot be changed in any way. Depending on how your study configured Onyx, it may or may not be possible to export the data. See Closing an Interview.
- `Cancelled` — This status means that the participant was withdrawn from the study. A staff member cancelled the interview. A cancelled interview cannot be restarted. See Cancelling an Interview.

> Your study customized what Onyx does when an interview is cancelled or closed. It may delete the participant or may keep some or all of the data so it can be exported.

Procedures

This section contains procedures related to viewing participants, their registration information, and their interviews.
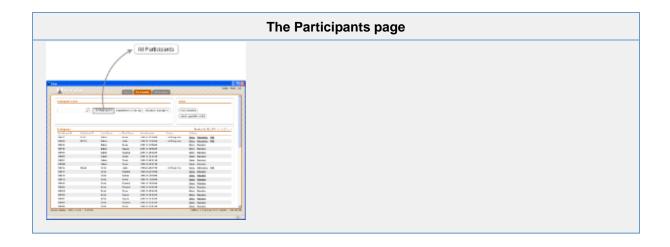
Displaying a List of All Participants

You can view a list of all the participants in the Onyx database: past and future participants, as well as those whose interviews are scheduled for the current date---today. The list does not include participants that have been purged from the Onyx database. See Data Purge.

Procedure

1. If the `Participants` page is not displayed, click the `Participants` tab.
2. Click the `All Participants` button. The `Participants` page is updated to show all participants in the Onyx database.

The Participants page

 Displaying a List of Today's Interviews

You can view a list of all the participants whose interviews are scheduled for the current date---today.

Procedure

1. If the `Participants` page is not displayed, click the `Participants` tab.
2. Click the `Appointments of the day` button. The `Participants` list is updated to show all participants who have interviews today.

The Participants page

 Displaying a List of Participants Currently Being Interviewed

You can view a list of all the participants whose interviews have the status `In Progress`. This status does not necessarily mean that the participant is seated with a staff member and that data is being collected from the participant. `In Progress` simply means that the particicipant's interview was started and that it is not finished yet.

Procedure

1. If the `Participants` page is not displayed, click the `Participants` tab.# Click the `Interviews in progress` button. The `Participants` list is updated to show all participants whose interviews have the status `In Progress`.

The Participants page
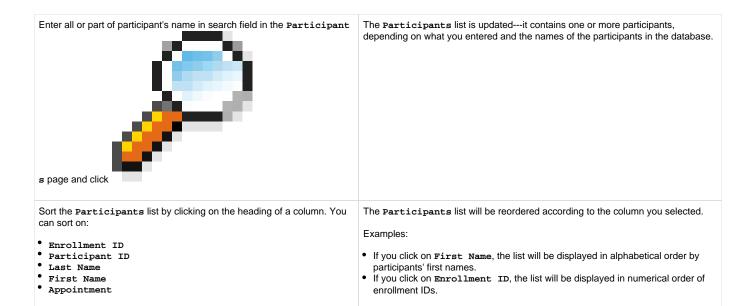
If you see a 🔒 icon at the end of the row for the participant, this indicates that an Onyx user has a lock on the interview. See Determining who has a lock on a participant.

Finding a Participant

Onyx gives you several ways to find participants. this figure summarizes the techniques you can use to search for participants.

Technique	Result
Scan participant's barcode (or enter participant ID manually) in the `Home` page and click `Go to interview`	The participant's `Interview` page opens.
Scan participant barcode (or enter participant ID in the search field) in the `Participants` page and click	The `Participants` list is updated and will include only one participant---the one with the scanned barcode
Enter participant ID or enrollment ID in search field on `Participants` page and click	The `Participants` list will be updated and will include only one participant---the one with the ID that you entered.

Enter all or part of participant's name in search field in the `Participant` **s** page and click	The `Participants` list is updated---it contains one or more participants, depending on what you entered and the names of the participants in the database.
Sort the `Participants` list by clicking on the heading of a column. You can sort on: - `Enrollment ID` - `Participant ID` - `Last Name` - `First Name` - `Appointment`	The `Participants` list will be reordered according to the column you selected. Examples: - If you click on `First Name`, the list will be displayed in alphabetical order by participants' first names. - If you click on `Enrollment ID`, the list will be displayed in numerical order of enrollment IDs.

If you cannot find a particular person who has arrived for an interview, a couple of explanations are possible:

- The appointment list may not be up-to-date.
- The person does not actually have an interview.

In either case, contact your participant manager who can update the appointment list or enroll a volunteer if your study accepts them.

Viewing a Participant's Registration Information
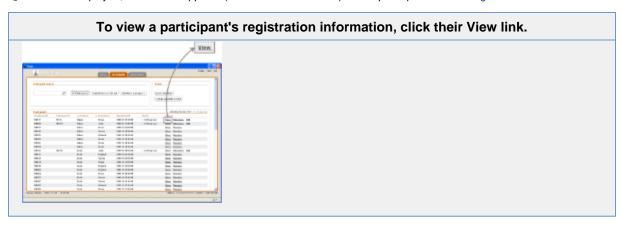
After a participant has been added to the Onyx database (either through an update of the appointment list or when they were enrolled as a volunteer), you can view their registraton information.
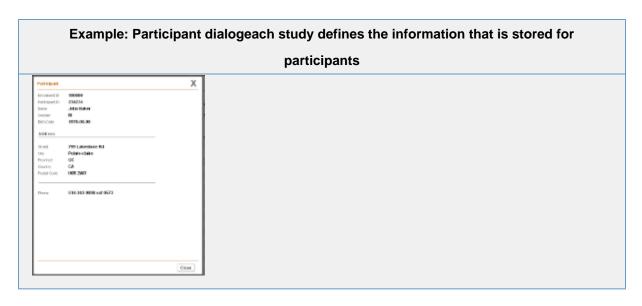
Procedure

1. If the `Participants` page is not displayed, click the `Participants` tab.
2. Find the participant whose information you want to view. If you need help finding the participant, see Finding a Participant. When the `Participants` list is displayed, a `View` link appears (in the `Actions` column) for the participant. See this figure.

To view a participant's registration information, click their View link.

3. Select the `View` link for the participant. The `Participant` dialog is displayed. See this figure.

> Each study defines the information that is stored for its participants. You may not see the same fields as those shown in this figure.

Example: Participant dialogeach study defines the information that is stored for participants
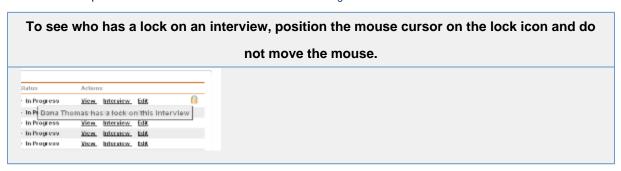
4. When you have finished viewing the information, select the `Close` button. The `Participant` dialog closes.

Determining Who Has a Lock on an Interview

If a lock icon ( ) is displayed at the right end of the row for a participant in the `Participants` list, an Onyx user has a lock on the interview, they are probably in the middle of an interview stage. You might even have a lock on the interview yourself, if you logged in on another workstation and worked on the interview from there. You can find out who has a lock on the interview as follows:

1. Move the mouse until the arrow cursor is on the lock icon and do not move the mouse for a few seconds. A small popup message will show the name of the person who has the lock on the interview. See this figure.

To see who has a lock on an interview, position the mouse cursor on the lock icon and do not move the mouse.

If necessary, your participant manager can unlock the interview. See Unlocking an interview.

Managing an Interview

Managing an Interview

Interviewing a participant in Onyx is a multiple stage process. You enter and control the interview from one place, the `Interview` page, which gives you access to all the stages of the interview for a particular participant. Interview stages include everything from signing a consent form to concluding the interview. Most stages are questionnaires, collection of biospecimens, and physical measurements.

Due to Onyx's high degree of customizability, this chapter cannot cover the actual stages defined by your study. Instead, this chapter covers the features and procedures that allow you to start an interview, to work your way through the stages of the interview, and to end an interview a safe way in various circumstances. By *safe*, we mean, so that you do not lose any data that you have collected.

This chapter covers these topics:

- The Interview page and its important features
- Key concepts related to the interview workflow
- The interview workflow broken down into steps
- Routine procedures at the interview level
- Routine procedures at the stage level

- [Procedures for unusual situations](#) (such as a participant withdrawing from the study or leaving before the interview is complete)

The Interview Page

When you find a participant and click their `Interview` link, the `Interview` page is displayed. This page gives you an overview of a participant's interview and shows the status of each interview stage. It is from this page that you start working on each stage of an interview and to which you are returned after a stage is complete. [this figure](#) shows the `Interview` page and points out some of its more important features.

The Interview page

Here are the important features of the `Interview` page:

- `Interview Exit buttons`. The `Exit X` and the `Exit this interview` button both close the `Interview` page properly---they release the lock that was put on the interview when you entered the `Interview` page. See [Obtaining a Lock on an Interview](#).
  You can select either `Exit` button when you need to leave the `Interview` page before the interview is complete. They can be used at any time and do not affect the status of the interview.
  Some examples of when you would exit an interview (using either of these buttons):
    - After you complete a stage and need to take a break
    - After you view the state of the participant's interview for information purposes
    - When you have finished the stages for which you are responsible and are ready to pass the participant over to another staff member
      The two `Exit` buttons function in exactly the same way---there are two of them for convenience sake.
- The **Onyx Exit button** (the `X` in the upper right corner of the Onyx window---see [this figure](#)) is not the same as the **Interview Exit** buttons. If you select the **Onyx Exit button** while in the `Interview` page, Onyx will close immediately without releasing your lock on the interview. The next time anyone (including you) looks for the participant in the `Participants` list, a lock will be displayed on the interview. A participant manager will have to unlock the interview. See [Unlocking an Interview](#).
- `Close` button — You select this button if you need to end the interview early (before all stages are complete), but do not need to withdraw the participant from the study.
  CAUTION: If you select `Close`, you will not be able to resume the interview.

  > Each study defines what will be done with the data acquired to the point of stopping an interview in this way.

- `Print Reports` button — Each study can define one or more reports that can be printed and given to the participant. You can select this button after any stage of an interview. It allows you to print reports as you go along rather than waiting until all stages are complete.
- `Cancel` — You select this button if you need to cancel the interview and withdraw the participant from the study.
  CAUTION: If you select `Cancel`, you will not be able to resume the interview.

  > Each study defines what will be done with the data acquired to the point of cancelling the interview.

- `Log` panel (in the Interview section of the page) — You use the buttons in this panel to view system messages and user comments for all the stages of the interview. You can also enter general comments about the interview.
- `Log` column (in the Stages section of the page) — You click the icons in this column to view logs (system messages and user comments) about particular stages of the interview. You can also view a list of user comments only for a particular stage.
- 🔍 (`View` button) — The `Interview` page contains several of these buttons: one for the interview in general, one for each interview stage that has been started. When you want to view a particular log, click its 🔍 button.
- 🗒 (`Comments` button) — Onyx users can add comments about the interview in general and about individual interview stages. The `Interview` page contains one or more of these buttons: one for the interview in general and one for each interview stage that has user comments. When you want to view user comments, click the appropriate 🗒 button.

- **Status** column (in the Stages section of the page) indicates the status of each stage of the interview. See Stage Status.
- **Actions** column (in the Stages section of the page) indicates the actions that are available for each stage of the interview. See Stage Actions.

Key Concepts

This section contains key concepts related to working your way through the stages of an interview.

Obtaining a Lock on an Interview

When you enter the **Interview** page of a participant, you obtain a lock on the interview. No other user can access the particpant's Interview page until you exit the **Interview** page properly (by selecting an **Interview Exit** button--- see Interview Exit buttons). If Onyx crashes or if you exit the **Interview** page improperly (by selecting the **Onyx Exit** button---see this figure), a participant manager will have to unlock the interview. See Unlocking an Interview.

Stage Status

An interview stage passes through various statuses from "ready" to "completed". Each type of stage (consent, questionnaire, physical measurement, specimen collection, interview conclusion) follows a particular path through the statuses. The **Actions** that are available depend on the **status** of the stage

> Each study defines the conditions that are required for a stage's status to change. A study can customize the names of the statuses that appear in the **Interview** page.

Stage Actions

The **Actions** column lists the actions that are available to you for a particular stage of the interview. The Actions depend on the status of the stage.

> Each study defines the actions that are available when a stage has a certain status. A study can also customize the names of the actions that appear in the **Interview** page.

Despite all the customization that Onyx allows, only five stage actions are possible: By default, the actions have the names indicated in this list.

- **Start** — You select this action to start a stage.
- **Skip** — You select this action to skip a stage. Some examples of why you might need to skip a stage: the stage is **Not ready**; it is more convenient to do the stage later; another staff member is responsible for the stage.
- **Reinstate** — You select this action to redo a stage that was skipped.
- **Cancel** — You can cancel a stage that is contraindicated or not applicable for some reason.
- **Review** — You can review a stage that was contraindicated in order to determine if the contraindication is appropriate or, if the answers that caused it should be corrected so that the contraindication will be removed.

Closing vs. Cancelling an Interview

Closing an interview means ending the interview early before all stages have been completed (or skipped, if your study permits). The participant is not withdrawn from the study.

Cancelling an interview means withdrawing the participant from the study. The decision to withdraw a participant may come from the participant or a staff member.

CAUTION: If you select the **Close** or **Cancel** button in the **Interview** page, you will not be able to continue the interview with the participant later. If you want to leave the **Interview** page temporarily and continue the interview later, you need to exit the **Interview** page. See this figure.

> Each study defines what will be done with the data acquired to the point of closing or cancelling an interview.
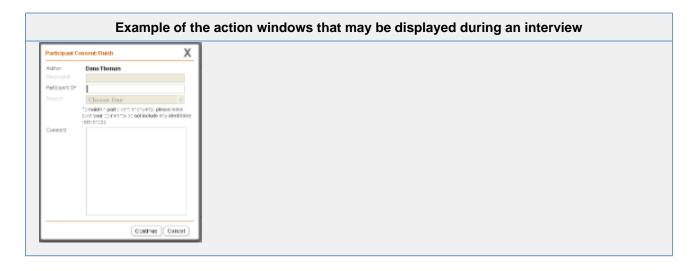
🪶 Reviewing vs. Reinstating a Stage

You can *review* a stage if its status is `Contraindicated`. If you review the stage and change the condition (for example, the answer to a particular question) that caused the stage to be contraindicated, you can actually change the status of the stage from `Contraindicated` to `Ready`.

You can *reinstate* a stage that was skipped. When you reinstate a stage, its status will change to `Ready` and it will be possible to `start` the stage again.

🪶 Action Windows

Depending on how your study customized Onyx, Onyx may display an *action window* when you select an `Action` in the `Interview` page. These windows usually require you to enter your password and the Participant ID (you can scan the barcode or enter it manually). An action window may contain other fields depending on the action that you selected and how your study customized the window. The purpose of action windows is to ensure the integrity of the data being collected. When you enter your password, you confirm that it was actually you who carried out the action. When you enter the `Participant ID`, you confirm that the data is indeed being collected for the correct participant. this figure shows an example action window.

Example of the action windows that may be displayed during an interview

🪶 Stage Navigation Buttons: Previous, Next, Finish

When you are in a stage, you will see various buttons at the bottom of the page. These buttons allow you to move back and forth through the pages of the stage and to finish the stage. See this figure.

By default, the buttons are named as shown in this figure. Your study may have changed the names on the buttons. .

The buttons that allow you to move through and finish the pages of an interview stage

You use the buttons as follows:

- `Next` button — Select it to display the next page in the stage.
- `Previous` button — Select it to display the previous page in the stage if you need to check or change something in that page.
- `Finish` button — This is only displayed when you are on the last page of a stage. You select it when you are ready to complete the stage.
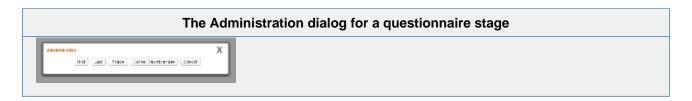
You can also use hot keys to navigate through the pages. See Hot Keys.

Administration Button and Administration Dialog

The `Administration` button is always displayed at the bottom of the page during an interview stage. When you select the `Administration` b utton, the `Administration` dialog is displayed. See this figure. You access this dialog when you need to pause or cancel the stage.

For a questionnaire stage, the dialog includes `First` and `Last` buttons. These buttons allow you to jump directly to the first and last pages that need your input. If no pages require your input, they jump to the first page or last page of the questionnaire.

The Administration dialog for a questionnaire stage

Hot Keys

To make the task of completing long questionnaires easier, Onyx allows you to use certain keyboard keys in a special way. When you are working in a stage, you can use these *hot keys* instead of having to use the mouse to select buttons. this figure lists the hot keys available during interview stages.

Onscreen Button	Hot Key	Effect	Comments
`Next`	The **+** key on your keyboard	Displays next page of the stage	Don't press **Shift**. Works for all kinds of stages, but not on the last page of a stage.
`Previous`	The **---** key on your keyboard	Displays previous page of the stage	Don't press **Shift**. Works for all kinds of stages
Finish	The **F** key on your keyboard	Finishes the stage. The stage closes and you are returned to the `Interview` page.	Don't press **Shift**. Works for all kinds of stages, but only on the last page of a stage.
`Administration`	The **A** key on your keyboard	Displays the `Administration` dialog.	Don't press Shift. If a text field is selected (the focus is on the text field), this hot key will not work. This is so you will be able to enter the letter "a" in the text field.
First	The **Home** key on your keyboard	Jumps to the first page that needs your input. If no page requires your input, it jumps to the first page of the questionnaire.	Only works in questionnaire stages
Last	The **End** key on your keyboard	Jumps to the last page that needs your input. If no page requires your input, it jumps to the last page of the questionnaire.	Only works in questionnaire stages
The **x** button in the corner of a dialog	The **Esc** key on your keyboard	Closes any dialog or small window that is displayed on top of a stage page.	Does not work on stage pages

Log

In the `Interview` page, a log contains a list of system messages and user comments. A log can either be about the interview as a whole, or about a particular stage of the interview. The list is in chronological order---the oldest entry is at the top, and the most recent entry is at the top bottom.

Comment

In the `Interview` page, a comment list only contains comments added by users. Depending on which button you select, the list of comments will either be about the interview as a whole, or about a particular stage of the interview. The list is in chronological order---the oldest entry is at the top, and the most recent entry is at the top bottom.

The Interview Workflow

Due to Onyx's highly customizable nature, we cannot tell you which stages you will see when you enter the `Interview` page or the exact names of the stage statuses and stage actions that you will see, but we can tell you--more or less--what you will do each time you

enter the `Interview` page.

Basic Workflow

The basic workflow of an interview is as follows:

1. **Enter the Interview page**. You enter the `Interview` page. You look at the list of stages and determine the next stage that must be done. Consent is usually the first stage. If it hasn't been obtained, you must obtain it.
2. **Start a Stage**. You start a stage. You would usually do the next stage that is "ready" and has a "start" action available for it.

> Depending on how your study customized Onyx, an action window may be displayed when you select a stage action (start, pause, resume, and so on). If an action window is displayed, you must complete it to confirm your identity and the identity of the participant, and, in some cases, to enter a comment explaining why the action was necessary.

1. **Work your Way through the Stage**. When you are inside a stage, you follow the directions displayed onscreen to collect the data required for the stage. You step through the pages of the stage by selecting the `Next` and `Previous` buttons. If you need to pause or cancel the stage, you select the `Administration` button.
2. **Finish the Stage**. When you have completed all the pages of a stage, the `Finish` button is displayed. You select the `Finish` button. Depending on how your study customized Onyx, you may have to complete an `Action` dialog. After that, you can start working on another stage or you can exit the `Interview` page (for example, you may need to take a break or pass the interview over to another staff member).
3. **Do all data-collecting stages**. You would repeat B, C, and D for each stage.
4. **Conclude the Interview**. Studies usually define a conclusion stage. If your study defined a conclusion stage, you would do it after all data-collecting stages have been completed (or skipped if your study allows this for certain stages). You complete a conclusion stage in the usual fashion: start, next, next, next, finish. After, this all the stages will have the status "completed" (or "skipped"). The interview is closed. You cannot make any changes to the data.
5. **Exit the Interview**. You exit the `Interview` page in the usual way, by selecting either of the **Interview Exit** buttons.

Common Alternative Steps

Here are some alternative steps that you will probably use fairly often:

- **Pause the Stage**. If you need to pause a stage, you select the `Administration` button and then the `Pause` button in the `Administration` dialog. You will have to fill out an `Action` dialog again to confirm your identity and the participant's identity. After that, you can exit the `Interview` page. Later, when you or another Onyx user wants to continue working on that stage, you re-enter the `Interview` page, and you will be able to resume the stage.
- **Exit the Interview**. If you leave a stage by pausing or finishing it. You can exit the `Interview` page, if necessary, by selecting either of the **Interview Exit** buttons. You will be able to re-enter the stage later, if you need to complete it or modify it.
- **Re-enter the Interview page of a participant**. You can always re-enter the `Interview` page of a participant immediately or afterward, by selecting the `Interview` action for the participant in the `Participants` page. You might want to do this in order to modify the data collected during a stage or to check how the interview went: were all stages completed, what were the user comments and system messages, how long a stage took, and so on.

Occasional Alternative Steps

Here are a few alternative steps that you may use occasionally:

- **Skip a Stage**. If you need to skip a stage (usually because of a technical problem like an instrument not being available for a physical measurement), you can select "skip" in the `Actions` column of the `Interview` page. You may or may not be able to complete the interview if it has skipped stages---it depends on how your study defined the conditions for completing the study.
- **Modify a Stage**. Occasionally, you may need to change a participant's answer to a question, or some other data collected during a stage. To do this, you modify the stage which opens the stage again. You can step through the pages of the stage and make the necessary changes.
- **Delete a Stage**. Occasionally, you may want to get rid of the data collected during a stage and start the stage over again. To do this, you delete the stage. Then you will be able to start it again.
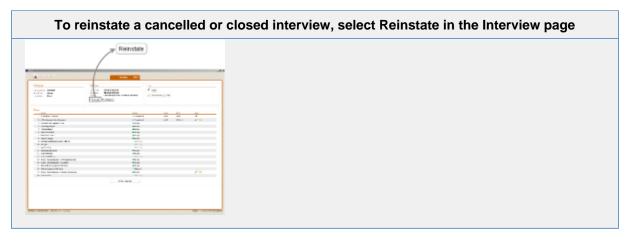
Rare Alternative Steps

These are alternative steps that you will use rarely:

- **Cancel the Interview**. If the participant decides they don't want to be part of the study, you can withdraw them by cancelling the interview. In this case, you select the `Cancel` button in the `Interview` page. You will have to complete an `Action` dialog as usual. Your study will have decided what it wants to do with the data acquired up to the point of cancelling an interview. Follow the onscreen directions.
- **Close the Interview**. If the participant has done part of the interview, but must leave early for some reason, you can end the interview early, without actually withdrawing the participant. In this case, you select the `Close` button in the `Interview` page. You will to complete an `Action` dialog. Your study will have decided whether it will keep or delete the data acquired up to this point. Follow the onscreen

directions.

- **Reinstate the Interview**. If by mistake you cancel or close the interview, you can restore the interview to its earlier state by selecting the `Reinstate` button in the `Interview` page. See this figure

To reinstate a cancelled or closed interview, select Reinstate in the Interview page

Routine Procedures at the Interview Level

These are procedures that relate to the interview as a whole:

- Entering the Interview page
- Exiting the Interview page
- Concluding the interview normally
- Printing reports for the participant
- Adding comments at the interview level
- Viewing comments at the interview level
- Viewing the log for the entire interview

Entering the Interview Page

After a participant is received or enrolled, you can enter their `Interview` page as explained below. If you exit the `Interview` page, you can also reopen it in the same way.

Procedure

1. Find the participant you want to interview. See Finding a Participant if you need help locating the participant. The participant should be displayed in the `Participants` page.
2. Select the participant's `Interview` link. The `Interview` page is displayed. You can now start any stage whose status is `Ready`.
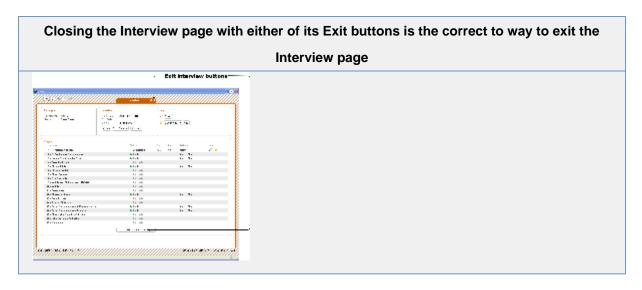
Exiting the `Interview` Page

As explained earlier in this chapter (see Exit buttons), you may need to exit the `Interview` window for any number of reasons. Exiting the `Interview` page does not stop the interview prematurely or withdraw the participant from the study. You and other Onyx users will be able to come back into the `Interview` page later. If the interview was not already completed, you will be able to continue it.

Procedure

1. If you are in the middle of an interview stage, pause it as explained in Pausing a Stage. The `Interview` page should be displayed.

Closing the Interview page with either of its Exit buttons is the correct to way to exit the Interview page

2. Select either of the interview **Exit** buttons in the **Interview** page (but not the X button in the upper right corner of the window). See thi s figure. The **Interview** page closes and the **Participants** page is displayed.

Concluding the Interview Normally

When all the stages of a study have been completed (or skipped if your study allows skipping of certain stages), the interview is finished. You can print reports for the participants, if your study has defined any. See Printing reports.

Printing Reports for Participants

Your study may have defined some reports that you can print and give to the participant for their information. You can print the report for a certain stage any time after the stage is complete.

Prerequisites

- A printer must be set up and ready for use from your workstation. If you have questions about the printer setup, speak to your system administrator.
- The participant's **Interview** page must be displayed. See Entering the Interview Page.

Procedure

1. Select the **Print Report(s)** button. The **Report** dialog is displayed. If the consent form is not available for printing, the participant may have signed a manual consent form.
2. Check the report(s) you wish to print, choose a language for each report, and select the **Print Reports** button.
3. If a **Compose Fax** dialog is displayed, select the **Exit** button.
4. When a dialog informs you that the reports have been sent to the printer, close the dialog. The **Interview** page is visible again.
5. Collect the reports from the printer.

Adding Comments about the Interview

When you are in the **Interview** page, you can add comments about the interview as a whole. The comments are added to the interview log (which also includes system messages) and to the comments list for the interview (which also includes comments about individual stages that are collected from the various action windows that are displayed during the course of the interview). See this figure

<table>
<tr><td>**To add a comment about the interview as a whole, select Add in the Log section of the Interview page**</td></tr>
<tr><td></td></tr>
</table>

Prerequisites

The participant's `Interview` page must be displayed. See Entering the Interview Page.

Procedure

1. Select the `Add` button in the `Log` section of the `Interview` page. A dialog is displayed.
2. Enter your comment in the dialog box.
3. If you decide you do not want to save the comment, select `Cancel`. The dialog closes and the `Interview` page is visible again.
4. If you want to save the comment, select `Save`. The dialog closes and the `Interview` page is visible again.
5. If you want to check that your comment was added, select the `Comments` button in the `Log` section of the `Interview` page. A `Log` window is displayed. It lists the comments on the interview, including the one you just entered.
6. When you have finished viewing the log, select the `Close` button. The `Log` window closed and the `Interview` page is visible again.

Viewing All User Comments about the Interview

You can view a list of all the comments made by users during the interview. Some comments come from the action dialogs about individual stages of the interview. Some of the comments may have been entered in the `Interview` page.

Prerequisites

The participant's `Interview` page must be displayed. See Entering the Interview Page.

Procedure

1. Select the `Comments` button in the `Log` section of the `Interview` page. A `Log` window is displayed.
2. View the comments and, if you want to add another one, select `Add`. See Adding Comments.
3. When you have finished viewing the comments, select `Close` in the `Log` page. The dialog closes and the `Interview` page is visible again.

Viewing the Interview Log

You can view a log of all the user comments and system messages for the entire interview. The user comments can come from the action dialogs about individual stages of the interview. or may have been entered in the `Interview` page. The system messages are mostly about actions taken during the different stages of the interview, but may also be about errors.

Prerequisites

The participant's `Interview` page must be displayed. See Entering the Interview Page.

Procedure

1. Select the `View` button in the `Log` section of the `Interview` page. A `Log` window is displayed.

2. View the log. You will see:
    - User comments — Indicated by a 🗨 icon and the word `Comment` in the `Action` column.
    - System messages — Indicated by an action (`Start`, `Finish`, and so on) in the `Action` column. The system generates a message whenever a user carries out an action in the `Interview` page. The system also generates messages if an error occurs.
3. If you want to add a comment, select the `Add` button. See Adding Comments.
4. When you've finished viewing the log, select `Close` in the `Log` page. The dialog is closed and the `Interview` page is visible again.

Routine Procedures at the Stage Level
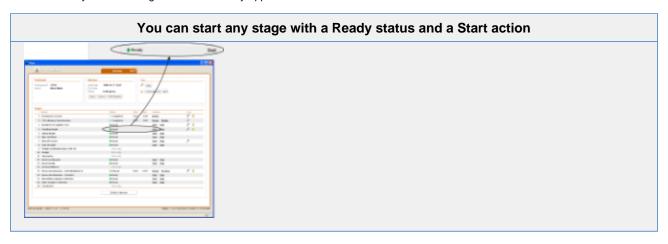
Routine procedures at the Stage Level (used during an interview stage---you initiate them from the Administration dialog)

- Starting a stage
- Pausing a stage
- Resuming a stage (that was paused)
- Cancelling a stage
- Finishing a stage
- Skipping a stage
- Reinstating a stage (that was skipped)
- Modifying a stage (that was already completed)
- Deleting a stage (that was already completed)
- Viewing the log for a stage
- Viewing comments about a stage

Starting a Stage

You can start any stage whose status is `Ready` and which has a `Start` link in the `Actions` column. See this figure. To avoid problems, it is recommended that you do the stages in the order they appear in the list.

You can start any stage with a Ready status and a Start action

Procedure

1. Select the stage's `Start` link.
2. If an action window for starting the stage is displayed, complete it. The action window closes and the stage window is displayed.
3. Complete the stage windows following the on-screen directions.

Finishing a Stage

When you are at the end of a stage, the `Finish` button is displayed in the lower right corner of the page.

Procedure

1. Select the `Finish` button.
2. If an action window for finishing the stage is displayed, complete it. The action window closes.

The `Interview` page is displayed. The status of the stage is now `Completed`. You can start another stage if there are any left to complete.

Pausing a Stage

When you are in the middle of a stage, you may need to pause the stage. You can do so when the `Administration` button is displayed in the lower left corner of the stage window. Afterwards, you (or another Onyx user) will be able to resume the stage. If the stage is a questionnaire, the questionnaire will resume after the last completed question.

Procedure

1. Select the `Administration` button. The `Administration` dialog is displayed. It includes a `Pause` button.
2. Select the `Pause` button.
3. If an action window for pausing the stage is displayed, complete it. The action window closes.

When the `Interview` page is displayed again, you can start another stage if any more need to be completed.

Resuming a Stage

If you (or another Onyx user) paused a stage, you can continue the stage as explained in this section. If the stage was a questionnaire, the questionnaire will resume after the last completed question.

Procedure

1. If you are not in the `Interview` page for the participant whose interview you want to resume, you may need to do some or all of the following to get there:
   - If your Onyx session expired, you must log in again. See Logging In.
   - If necessary, find the participant in the most convenient way for you. See Finding a Participant.
   - If the participant is listed in the `Participants` page, select their `Interview` link.
2. When you are in the participant's `Interview` page, select the `Resume` action for the stage that you wish to resume.
3. If an action window for resuming the stage is displayed, complete it. The action window closes.
4. When the stage page is displayed, you continue the stage where it had been paused. You can use hot keys to move through the pages more quickly. See Hot Keys.
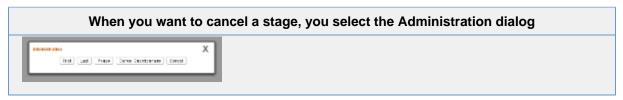
Cancelling a Stage

After you start a stage, you may decide not to proceed with it at the current time. When you are in the middle of the stage pages, you can cancel the stage as explained in this section.

Procedure

1. Select the `Administration` button. The `Administration` dialog is displayed. Depending on the type of stage, the dialog will include a button labelled `Cancel Questionnaire` or `Cancel Measurement` or `Cancel Sample Collection`.

When you want to cancel a stage, you select the Administration dialog

2. Select the appropriate cancel button (`Cancel Questionnaire` or `Cancel Measurement` or `Cancel Sample Collection`). Do not select the button labelled just `Cancel`--- it just closes the `Administration` dialog and puts you back in the stage.
3. If a dialog prompts you to confirm that you want to cancel the stage, select `Yes`.
4. If an action window for cancelling the stage is displayed, complete it. The action window closes.

When the `Interview` page is displayed again, you can restart the stage or start another stage.

Skipping a Stage

If you need to skip a stage, you can do so as explained in this section. If the stage you are skipping turns out to be necessary for another stage, you will find out when you start the other stage.

Prerequisites

- The correct participant's `Interview` page must be displayed. See Entering the Interview Page.
- The `Skip` action is available for the stage you wish to skip.

Procedure

1. Select the `Skip` action.
2. If an action window for skipping the stage is displayed, complete it. The action window closes.

When the `Interview` page is displayed again, you can start another stage if any more need to be completed.

Reinstating a Stage

If you (or another Onyx user) skipped a stage, you can reinstate it as explained in this section. Reinstating a stage means restoring it to the `Ready` status so it can be started again.

Prerequisites

- Ensure that any problems have been resolved. Read the user comments or the log for the stage to find out why it was skipped (see Viewing Logs and Comments for a Stage).
- The correct participant's `Interview` page must be displayed. See Entering the Interview Page.
- The `Reinstate` action is available for the stage you wish to skip.

Procedure

1. Select the `Reinstate` action for the stage you wish to reinstate.
2. If an action window for reinstating the stage is displayed, complete it. The action window closes.
3. When the `Interview` page is displayed again, you can start another stage or restart the same stage immediately, if you wish.

Modifying a Stage

You may need to reopen a stage that was alredy completed in order to change the data that was saved for it. For example, you may need to correct an answer in a questionnaire stage or redo a measurement in a physical measurement stage. If you need to reopen a stage in order to change some of the data stored for it, you can do so as explained in this section. After you make the changes, the stage's status will become `Completed` again.

If the stage is a long one, it is better to modify the stage rather than deleting it and starting it over from scratch (compare with Deleting a Stage).

The consent stage cannot be modified. If you need to change it in some way, you must delete it and start it over again. See Deleting a Stage.

Prerequisites

- The correct participant's `Interview` page is displayed. See Entering the Interview Page.
- The `Status` of the stage you wish to delete is `Completed`.
- The `Modify` action is available for the stage.

Procedure

1. Select the `Modify` action for the stage you wish to modify.
2. If an action window for modifying the stage is displayed, complete it. The action window closes.
3. When the stage opens, it shows all data that was stored previously. Step through the stage page(s) and make any changes that are necessary. You can use hot keys to move through the pages more quickly. See Hot Keys.
4. When you arrive at the last page of the stage, select the `Finish` button.
5. If an action window for finishing the stage is displayed, complete it. The action window is closes.

When the `Interview` page is displayed again, you can start another stage if any more need to be completed.

Deleting a Stage

You may need to redo a stage that was alredy completed. If you need to start a stage over from scratch, you can delete the stage as explained in this section. After deleting the stage, you will be able to restart it.

Prerequisites

- The correct participant's `Interview` page is displayed. See Entering the Interview Page.
- The `Status` of the stage you wish to delete is `Completed`.
- The `Delete` action is available for the stage.

Procedure

1. Select the `Delete` action for the stage you wish to delete (and later restart).
2. If an action window for deleting the stage is displayed, complete it. The action window closes.

3. When the `Interview` page is displayed again, you can start another stage or restart the same stage immediately, if you wish

Viewing Comments about a Stage of the Interview

Prerequisites

The participant's `Interview` page must be displayed. See Entering the Interview Page.

Procedure

1. Select the 🗒 icon at the right end of the line for the stage. A `Log` window listing all user comments about the stage is displayed.
2. View the user comments and, and if you'd like to view user comments for the entire interview, select the `Show All` button.
3. Select `Close` in the `Log` window. The window closes and the `Interview` page is visible again.

Viewing the Log about a Stage of the Interview

Prerequisites

The participant's `Interview` page must be displayed. See Entering the Interview Page.

Procedure

1. Select the 🔍 icon at the right end of the line for the stage. A `Log` window listing all system messages about the stage is displayed.
2. View the system messages. If you'd like to view the user comments and system messages for the entire interview, select the `Show All` button.
3. When you have finished viewing the log, select `Close` in the `Log` window. The window closes and the `Interview` page is visible again.

Procedures for Exceptional Situations
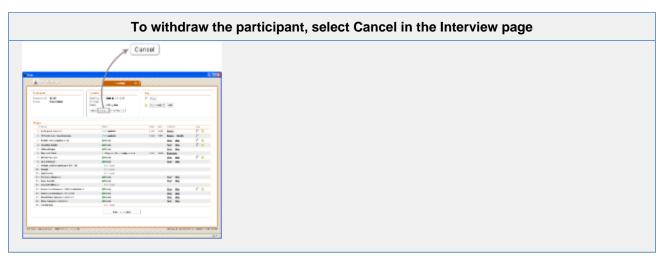
Occasionally, you may need to handle a unusual situation such as a participant withdrawing from the study or leaving before the interview is complete. This section includes the procedures for these situations:

- Cancelling the interview (withdrawing the participant)
- Closing the interview (ending the interview before completion)

Cancelling the Interview (withdrawing the participant)

If you need to cancel the interview and withdraw the participant from the study, you select the `Cancel` button in the `Interview` page. See this figure

To withdraw the participant, select Cancel in the Interview page

Prerequisites

The participant's `Interview` page must be displayed. See Entering the Interview Page.

Procedure
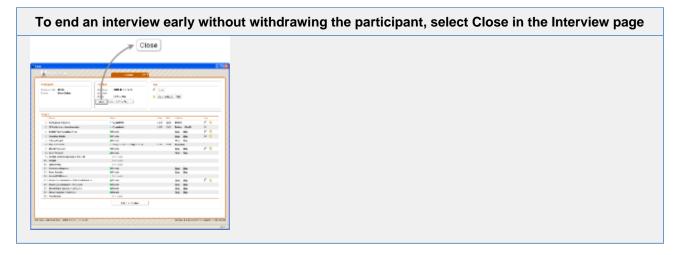
1. Select the `Cancel` button. A dialog is displayed. It warns you that the participant will be withdrawn from the study and asks you to confirm that you want to cancel the interview.
2. If you decide you do not want to cancel the interview, select `No`. The dialog closes and the `Interview` page is visible again.
3. If you want to go ahead and cancel the interview, select `Yes`. The dialog closes.
4. If an action window for cancelling the interview is displayed, complete it. The action window closes.

When the `Interview` page is displayed again, you will notice that the status of the interview is now `Cancelled`. You will not be able to start any stages. You can view logs and comments. When you exit the `Interview` page, you will be enter it again, but only to view the status of the various stages.

Closing the Interview (before completion)

If you need to end an interview early (before all stages are complete), and do not need to withdraw the participant from the study, you select the `Close` button in the `Interview` page. See this figure

To end an interview early without withdrawing the participant, select Close in the Interview page

Prerequisites

The participant's `Interview` page must be displayed. See Entering the Interview Page.

Procedure

1. Select the `Close` button. A dialog is displayed. It warns you that you will not be able to continue the interview and asks you to confirm that you want to close the interview.
2. If you decide you do not want to close the interview, select `No`. The dialog closes and the `Interview` page is visible again.
3. If you want to go ahead and close the interview, select `Yes`. The dialog closes.
4. If an action window for closing the interview is displayed, complete it. The action window closes.

When the `Interview` page is displayed again, you will notice that the status of the interview is now `Closed`. You will not be able to start any stages. You can view logs and comments. When you exit the `Interview` page, you will be enter it again, but only to view the status of the various stages.

Obtaining Participant Consent

Obtaining Participant Consent

For most studies, the first stage of an interview with a participant is a consent stage. During the consent stage, two things must happen: the participant must read and sign a consent form (or the participant may refuse to sign the form), and their consent (or refusal to consent must be registered in Onyx. Onyx allows studies to obtain and register consent in two ways: manually (using a paper form) and electronically (using an on-screen form and an electronic signature pad).

In Onyx, obtaining consent is a stage of the participant interview and can be controlled in the same way as other stages. The general procedures for controlling an interview stage apply to the consent stagesee this figure.

All Onyx interview stages, including the consent stage, are highly customizable. Each study writes the content of its consent form, and
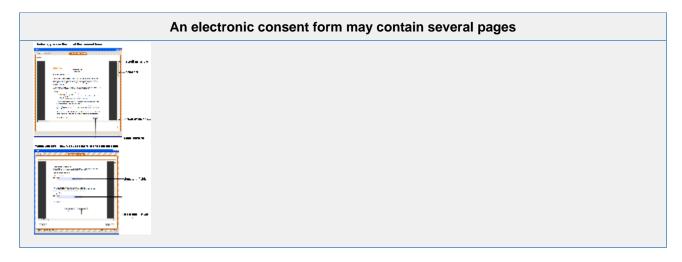
decides in which ways the participant will be allowed to read and sign the consent form.

This chapter covers the following topics:

- An electronic consent form and its important features
- Key concepts related to obtaining consent
- Procedures used to obtain consent

Electronic Consent Forms

In Onyx, an electronic consent form is a file in PDF format (Portable Document Format). An electronic consent form contains the text that was defined by the study and that the participant must agree to in order to participate. The form is displayed on-screen during the consent stage so that the participant can read and sign it. The example in this figure shows some of the features of electronic consent forms. A consent form can contain buttons that click or check to select options and fields that you must click in and complete. PDF files have many display options that you can access by clicking on the consent from with the right mouse button. For some pointers, see Tips for Viewing an Electronic Consent Form.

An electronic consent form may contain several pages

Key Concepts

This section contains key concepts related to obtaining participant consent.

Manual vs. Electronic Consent

Studies can customize how consent is obtained:

- Electronic Consent The participant reads the consent form on the workstation screen, then signs on an electronic signature pad, and the electronic signature is stored on the electronic consent form.
- Paper Consent The participant reads and signs a paper consent form, and a staff member makes a selection in Onyx that indicates that the participant signed a paper consent form.
- A study can customize Onyx to accept consent both electronically and manually

Studies customize the text in the consent form and the name of the consent stage in Onyx.

Electronic Signature Pad

An electronic device used to capture an electronic signature. See this figure. An electronic signature is any legally recognized electronic means that indicates that a person agrees to the contents of an electronic message. In this case, the electronic message is an electronic consent form.

An electronic signature pad

Pointers for Viewing an Electronic Consent Form

Before you do an electronic consent stage with a participant, try out the following techniques for viewing and completing the form:

If the form is too large to be displayed all at once:

Use the scrollbar and scroll arrows to move around the consent form. They are located to the right of the formsee this figure.

- To move down through the form, click the scroll-down arrow one or more times.
- To move up through the form, click the scroll-up arrow one or more times.
- To move up and through the form, click and drag on the scrollbar.

If the text is too large or too small:

- To make the text bigger, press and hold the **Ctrl** key and click the **+** key (the plus key) once or several times.
- To make the text smaller, press the **Ctrl** key and click the **-** key (the minus key) once or several times.

If you want to reset the form (to its original size and with all fields cleared):

- Click the `Previous` button and then the `Next` button.

Procedures

This section contains procedures used to obtain participant consent.

Obtaining Consent Manually

> If your study uses paper consent forms (or if it allows both paper and electronic consent forms), there will be a consent stage in your version of Onyx. This section explains how to record the result of the paper consent process in Onyx.

Prerequisites

- Instruct the participant to read through the form and to sign it when they are ready.
- While the participant is reading the form, ensure that the participant's `Interview` page is displayed. See Entering the Interview Page.
- Check that the `status` of the consent stage is `Ready`.

Procedure

You can start this procedure while the participant is reading the consent form so that you are ready to register whether or not they signed the consent form when they have finished reading.

1. Select the `start` link for the consent stage. An action window for starting the consent stage may be displayed.
2. If an action window is displayed, complete it. The action window closes and the stage window is displayed.
3. Enter your password, scan the participant's barcode (or enter their Participant ID manually) and select the `Continue` button.
4. If your study accepts electronic and paper consents, you will be prompted to select one or the other for the current participant. See this figure. Select paper consent.

If your study accepts consent in electronic and manual format, you will be prompted to select a format

5. Select the language of the form given to the participant.
6. When Onyx prompts you to select whether or not the participant has read and signed the consent, check if the participant has signed the consent form and select the appropriate response in Onyx.
7. When you have responded to all questions, select the `Administration` button. The `Administration` dialog is displayed
8. Select the `Finish` button. An action window for finishing the consent stage is displayed.
9. Scan or enter the Participant ID manually, then select the `Continue` in the action window. The action window closes and the `Intervie w` window is visible again.

 Obtaining Consent Electronically
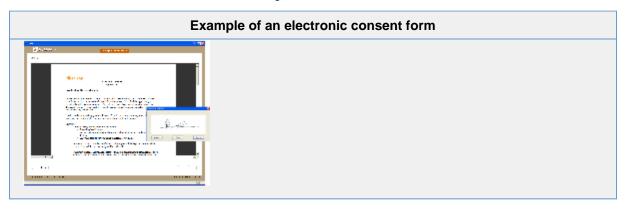
> If your study records consent electronically (or if it accepts both electronic and paper consent), there will be a consent stage in your version of Onyx. You obtain consent electronically as explained in this section.

Prerequisites

- An electronic signature pad must be be connected to the Onyx workstation.
- Ensure that the participant's `Interview` page is displayed. See Entering the Interview Page.
- Check that the `Status` of the consent stage is `Ready`.

Procedure

1. Select the `start` link for the consent stage. An action window for starting the consent stage may be displayed.
2. If an action window is displayed, complete it. The action window closes and the stage window is displayed.
3. Enter your password, scan the participant's barcode (or enter their Participant ID manually) and select the `Continue` button.
4. If your study accepts electronic and paper consents, you will be prompted to select one or the other for the current participant. See this figure. Select electronic consent, and follow the rest of the on-screen directions.If your study is set up for electronic consent, the forms you will see on-screen will be similar to those shown in this figure.

Example of an electronic consent form

5. Since the participant is completing the form on-screen, go over the following points:
   - Show them how to scroll up and down through the form, and adjust the text size if necessary. See Tips for Viewing an Electronic Consent Form.
   - Point out any buttons or fields where the participant must make a selection or an entry.
   - Advise them to read the form carefully and ask them to tell you when they are ready to sign it.
6. When the participant is ready to sign the form, show the participant how to use the electronic signature pad (if you have to co-sign the

form, you can demonstrate by signing your name in the field provided):
- Using the mouse, click in the signature field in the onscreen form.
- Use the stylus (attached to the signature pad) to sign your name in the window on the signature pad. As you write, the signature is displayed simultaneously on the signature pad and on the screen of the Onyx workstation.
- If the signature is well done, use the mouse to click `Accept` on the Onyx screen. If you don't select `Accept` within a few seconds, the signature will be cleared and you will have to sign again.
- When you are ready, select the `Accept` button at the bottom of the consent form.

7. Ask the participant to sign on the signature pad.
8. Ensure the participant has responded to any remaining questions on-screen.
9. If there is an `Accept` button, ask the participant to select it. An Action window may be displayed nextif so, go to this figure.
10. If the `Administration` button is displayed in the lower left corner of the page, select it. The `Administration` dialog is displayed.
11. If you don't want to complete the stage now, select the `Cancel Consent` button. An action window for cancelling consent is displayed.
12. If you want to complete the stage now, select the `Finish` button. An action window for finishing the consent stage is displayed.
13. Scan or enter the Participant ID manually, then select the `Continue` or `Cancel` button in the action window. The action window closes and the `Interview` window is visible again.

Completing Questionnaires

Completing Questionnaires

Questionnaires are an important component of participant interviews. An interview can include several questionnaires. In Onyx, each questionnaire is considered a stage of the participant interview, so the general procedures for controlling an interview stage (see this figure) apply to questionnaire stages.

> Questionnaires are highly customizable. Each study creates its own questionnaires from scratch and writes the directions to staff members that appear in the questionnaire.

Since your study has defined its own questionnaire stages, this guide cannot give you advice about how to answer particular questions in your study's questionnaire(s). Instead, this chapter covers the types of questions you will see in your questionnaires and gives you pointers about working on questionnaires in Onyx. Example questions are taken from an example Onyx web application. In particular, this chapter covers these topics:

- Key concepts related to questionnaires
- Pointers for completing assisted questionnaires with participants
- Example questions from assisted questionnaires
- Example questions from self-administered questionnaires

Key Concepts
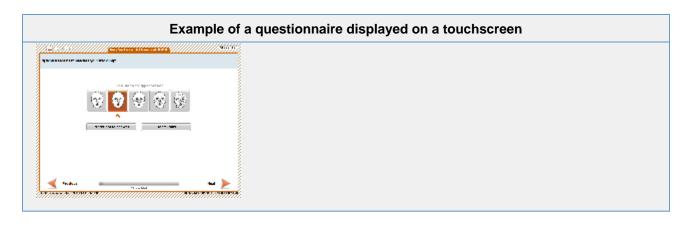
Touchscreen vs. Regular Screen

Onyx can display questionnaires on two different types of screen:

- **Touchscreen**. A touchscreen is a screen that allows you to select buttons by touching them on the screen with your finger. Your finger takes the place of the mouse on a regular screen. See this figure. Usually, a keyboard is not used with a touchscreen. If an Onyx questionnaire is configured for display on a touchscreen:
  - The buttons are larger (so they're easier to point at with a finger).
  - The questions are often multiple-choice (so you don't need a keyboard to enter an answer).
  - If you have to enter a number, a number pad is displayed on the touchscreen.
    Touchscreen display mode is especially suitable for self-administered questionnaires. The recommended resolution to use for a touchscreen is 1024 x 768.
- **Regular Screen**. By a regular screen, we mean the kind of screen used with most desktop computers and the screens of laptop computers. You select onscreen objects with a mouse (or a touchpad on a laptop). If an Onyx questionnaire is configured for display on a regular screen, the buttons are usually smaller than in touchscreen mode, and number pads are not displayed since a keyboard is available.
    Regular display mode is used for assisted questionnaires for assisted questionnaires.

Example of a questionnaire displayed on a touchscreen

Self-administered Questionnaires

A self-administered questionnaire is a questionnaire that the participant answers on their own, rather than being interviewed by a member of the study team. The questionnaire is usually displayed on a touchscreen. The participant answers the question by touching the appropriate answer on the touchscreen.

Assisted Questionnaires

An assisted questionnaire is that an interviewer *assists* the participant with. The interviewer sits with the participant, asks the questions that appear on the workstation screen, and enters the participant's answers in Onyx. The questionnaire is usually displayed on a regular workstation with a keyboard available. See Example Questions from Assisted Questionnaires.

Contraindication Questionnaires

Questionnaires can be used to establish whether a physical measurement or biospecimen stage is contraindicated. Contraindication questionnaires would typically be one of the earlier stages in the interview and, of course, have to be done before the stages that may subsequently be contraindicated. Contraindication questionnaires are just like any other questionnaire stage. You control them in the same way and answer the questions in the same way.

Answer Validation

Studies can validate the answers you enter in text fields. For example, if you have to enter a number such as the participant's age, Onyx may validate if it is in a certain range. If an answer you enter does not pass validation, you will see an error message that will usually give you some indication of how to correct the error.

Pointers for Completing Assisted Questionnaires

This section gives some general pointers for completing assisted questionnaires with participants the type that you read to the participant and enter their answers for them. Before reading these pointers, you should experiment with the assisted questionnaires that are part of your study. You can also look at the Example Questions from Assisted Questionnaires section of this chapter.

- Here are some keyboard shortcuts that you can use in any page of Onyxincluding questionnaires:
  - To increase the font size of text: Press **Ctrl** and **+** keys.
  - To decrease the font size of text: Press the **Ctrl** and **-** keys.
  - To jump forward between fields and/or buttons: Press the **Tab** key.
  - To jump backward between fields and/or buttons: Press the **Shift** and **Tab** keys.
- Always read the question clearly to the participant and let them know if they will be able to select just one answer or several answers.
  - If the answers are preceded by round buttons ⦿ , the participant can only choose one answer. See this figure:
  - If the answers are preceded by square checkboxes ☑ , the participant can choose several answers. See this figure.
  - If the answer includes a drop-down list ⬚ , you click and hold on the arrow to read the list of options, and then release on the participant's answer.
  - If the answer includes a text box ○ Other condition ⬚ , the participant may or may not have to supply an answerdepending on how your study defined it. For example, they might need to tell you about a condition they have that was not among the options they could choose from. One or more alternative answers are usually provided, in case the participant

cannot or does not want to provide the particular bit of information that you would usually enter in the text box.

- If the answer includes an auto-complete text box, you write the first letter of a word, Onyx predicts one or more possible words as choices.

- If the answer includes an audio recording answer, you click the red button and record the particpant answer.

00:00 / 00:00

- You can use the **Next** and **Previous** buttons to go to the next page of a questionnaire or to return to the previous page of a questionnaire. See this figure.
- You can also use hot keys to go forward and back through the pages of the questionnaire. See Hot Keys.
- With practice, you will learn how quickly you can proceed through a questionnaire. While learning, do not click the **Next** button too quickly, because some pages show only one question initially, and show additional question(s) after you answer the first question. See this figure. If you click **Next** too quickly, you will see an error message.
- Some questions that take the same type of answers are grouped together in a table (for example, medical history questionssee this figure). Be careful to select the correct answer for each question in the table.
- Always read on-screen directions carefully and follow them.
- If you need to pause or cancel the questionnaire, click the **Administration** button.

Example Questions from Assisted Questionnaires
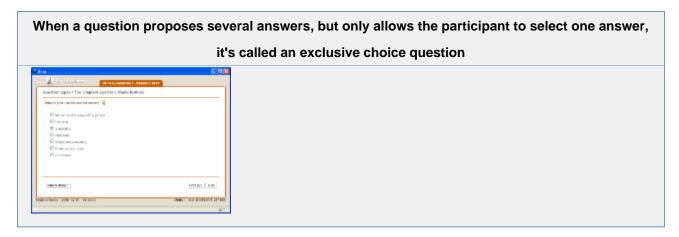
This section gives examples of the various types of questions and answers that you will see in your study's assisted questionnaire stages. They are not actual questions taken from your study's questionnaires.

 Exclusive Choice Questions
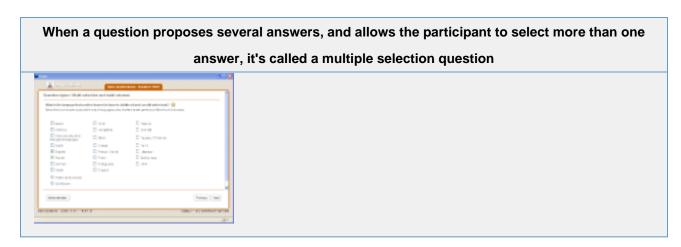
If a question proposes several answers, but only allows the participant to choose one answer, it will look like the question in this figure. You click on the round button (called a radio button) beside the answer that the participant chooses.

When a question proposes several answers, but only allows the participant to select one answer,
it's called an exclusive choice question

 Multiple Selection Questions

If a question proposes several answers, and allows the participant to choose more than one answer, it will look like the question in this figure. You click in the checkbox beside each answer that the participant chooses.

When a question proposes several answers, and allows the participant to select more than one answer, it's called a multiple selection question
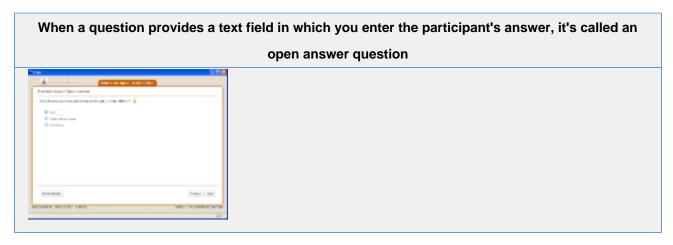
Open Answer Questions

Some questions provide a text field where you can enter the participant's reply. Such a question might look like the question in this figure. The question may also provide some default answers if the participant does not provide an answer that you can enter in the text field. You enter the answer that the participant says in the text field or select one of the other answers. If you enter some text in a text field, the round button beside that answer will automatically be selected.

If you must enter a date in a field, it may have a calendar beside it. You can click on the calendar and choose a date. The date you chose will be inserted in the field, and you can modify it, if necessary.

When a question provides a text field in which you enter the participant's answer, it's called an open answer question
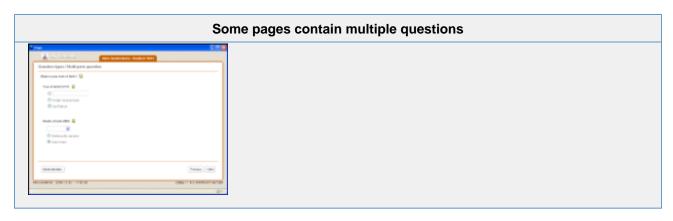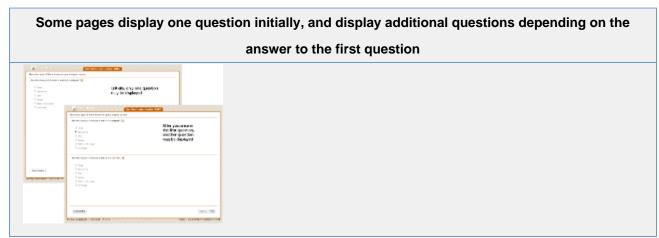
Multiple Questions on a Page

Some pages in a questionnaire contain several questions. Sometimes the page shows all the questions at once, as shown in this figure.
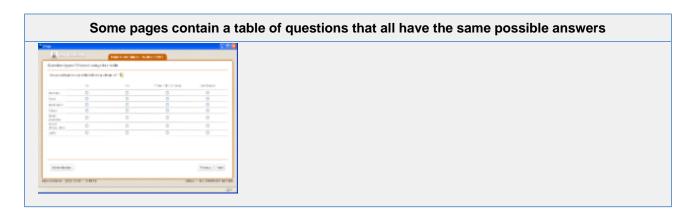
Sometimes, a page only shows one question initially, and after you enter the answer to the first question, another question is displayed as shown in this figure. For this type of page, if you select the **Next** button quickly, you may see an error message that tells you to reply to a question that was not displayed intially. Be sure to answer all questions on a page before selecting the **Next** button.

<table>
<tr><td align="center">**Some pages contain multiple questions**</td></tr>
<tr><td></td></tr>
</table>

<table>
<tr><td align="center">**Some pages display one question initially, and display additional questions depending on the answer to the first question**</td></tr>
<tr><td></td></tr>
</table>

▶ Questions with Shared Category Answers

Some questions can be grouped together into a table because the answers to all of them can be selected from the same set of options. this figure shows a page with a set of questions with shared category answers. Take your time in order to select the correct answer for each question in the table.

<table>
<tr><td align="center">**Some pages contain a table of questions that all have the same possible answers**</td></tr>
<tr><td></td></tr>
</table>

Example Questions from Self-Administered Questionnaires

This section gives examples of the types of questions that the participant will see in a self-administered questionnaire. They are not actual questions that you will see in a self-administered questionnaire in your version of Onyx. They are taken from an example Onyx web application, and serve to show the different types of question and answer formats that are possible.

▶ Exclusive Choice Questions on a Touchscreen

If a question proposes several answers, but only allows the participant to choose one answer, it will look like the question in this figure.

You must tell participants that they simply have to touch the answer that they want to choose.
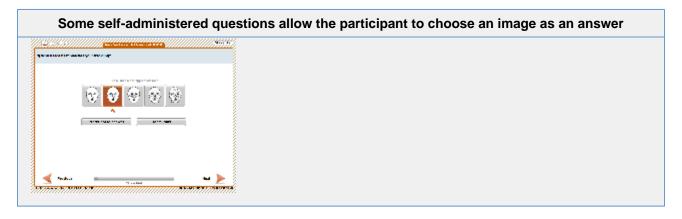
A touchscreen version of an exclusive choice question which proposes several answers, but only allows the participant to choose one

![Visual Choice icon] **Visual Choice Questions on a Touchscreen**

Some questions in a self-administered questionnaire allow the participant to choose their answer from a number of images. See this figure.

You must tell the participant to touch the image that best represents their answer to the question.

Some self-administered questions allow the participant to choose an image as an answer

![Number Pad icon] **Number Pad Questions on a Touchscreen**

Some questions in a self-administered questionnaire require the participant to use an onscreen number pad to enter a numeric value. See this figure.

You can give the participant the following explanation of how to use a number pad:

- To display the number pad, they must touch an onscreen button labelled **Press here**.
- They touch the keys on the number pad to tap out the number that answers the question.
- If they make a mistake, they can touch the **Clear** button.
- When the correct answer is displayed in the text box, they touch the **OK** button.

Some self-administered questions require the participant to enter numeric values on a number pad

 Questions in a Table on a Touchscreen

Some questions in a self-administered questionnaire may be grouped together into a table because the choice of answers can be used for all the questions. this figure shows a table with a set of questions with shared category answers.

You can give the participant the following explanation of how to complete a table:

- The column headings show the answers that they will choose from for each question in the table.
- Each row contains a question. They slide their finger along the row and press when they arrive at the best answer for the question. A checkmark will be displayed in the box they pressed on.
- If they want to change an answer, they simply press a different box. The check for the previous answer will be removed and a checkmark will be displayed in the box for their new answer.
- If they decide they want to start the table over from scratch, they press the `Clear` button. Any checkmarks that were in the table are removed.

Some self-administered questions are grouped together in a table because the choice of answers is the same for all of the questions

Collecting Physical Measurements

Collecting Physical Measurements

Physical measurements are often acquired for research studies. In Onyx, each physical measurement is considered a stage of the participant interview, so the general procedures for controlling an interview stage apply to physical measurement stages (see this figure).

All Onyx interview stages, including physical measurements, are highly customizable. Each study defines which physical measurements must be collected, the sequence in which they must be collected (if there are any dependencies among the measurements), and establishes a way to determine if any of the measurements are contraindicated. Since Onyx is so customizable, it allows studies to include directions to staff members right in the Onyx pages used to collect the physical measurements.

Due to the fact that each study's physical measurement stages are unique, this chapter covers typical physical measurement stages. Example stage pages are taken from the Onyx example-webapp. In particular, this chapter covers these topics:

- Key concepts related to using Onyx to record physical measurements
- Workflow of a physical measurement stage
- Example physical measurement stage

Key Concepts

Mechanical Instruments

Mechanical instruments do not have any electronic components and so do not need to be plugged in. Onyx cannot read measurements directly from these instruments. Some examples are:

- Grip strength dynamometer (used to measure grip strength)
- Tape measure (used to measure circumference of upper arm, waist, hips)
- Mechanical stadiometer (used to measure height)

For measurements that use mechanical instruments, you enter the values manually in Onyx. See this figure

Electronic Instruments

Electronic instruments are those that have electronic components. They usually have an on/off switch. They must be plugged in, or they are battery-powered, or perhaps both.

Here are some examples:

- Electrocardiogram
- Bone densitometer
- Electronic sphygmomanometer

Many electronic instruments can be connected directly to an Onyx workstation, so that Onyx can capture measurements automatically. If this is the case, they usually have their own software programme that you start from Onyx. See this figure. Some examples of electronic instruments that have their own software are:

- Minispir spirometer
- Sphygmocor CP (used to measure arterial stiffness)

Manual Entry of Measurements

Onyx permits manual entry of physical measurements. Manual entry is possible when measurements are done using:

- Mechanical instruments
- Electronic instruments that do not connect directly to an Onyx workstation
- Electronic instruments that are usually connected to Onyx---when the connection is not working

Automatic Capture of Measurements

Onyx can capture measurements automatically from instruments connected to the workstation. The capture is done in one of these ways, depending on the instrument:

- By launching the instrument's own computer programme
- By running a custom Onyx component that captures the measurements

Instruments Reserved for the Workstation

Each physical measurement stage requires a particular type of instrument, and each Onyx workstation must have a particular instrument of that type available to do the measurement. The Onyx `Workstation` page lists the instruments available on the workstation on which you are working. See Managing an Onyx Workstation.

When you start a physical measurement stage, Onyx checks the instrument list for the workstation to see if an instrument of the appropriate type is available. If Onyx determines that an appropriate instrument is available, you will be able to do the measurement. If Onyx does not find a suitable instrument for the measurement, you will not be allowed to proceed with the stage until an instrument is registered on the workstation. (see Registering an Instrument). Registering instruments makes your work easier (fewer barcodes to scan) and ensures that it will be possible to identify which instrument was used to take a particular measurement for a particular participant.
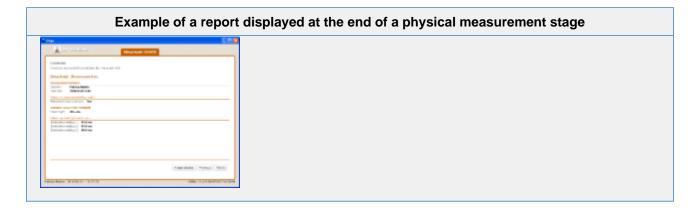
Multiple Measurements

> Depending on how the study defined a physical measurement stage, you may be allowed or required to enter more than one reading of a measurement.

Validation of Measurements

> A study can define the range of values that are acceptable for a physical measurement. If your study defined acceptable values for a particular physical measurement stage, Onyx will validate the values you enter for the measurement. If a value you enter is not acceptable, a dialog will inform you, and you will have to enter an acceptable value in order to be able to finish the stage. Onyx can also check for discrepancies among multiple measurements

On-screen Report of Measurements
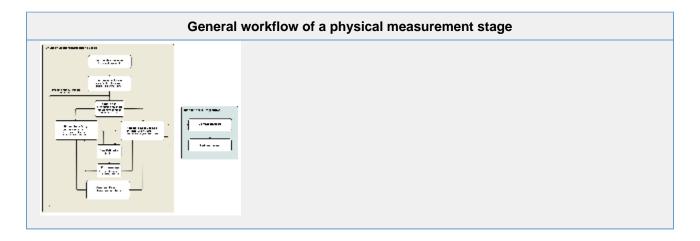
The last page of a physical measurement stage includes a report of the data collected during the stage. See this figure.

Example of a report displayed at the end of a physical measurement stage

Workflow of a Physical Measurement Stage

Since your study has defined its own physical measurement stages, this guide cannot give step-by-step procedures for taking physical measurements in your version of Onyx. Instead, this section describes the general workflow of a physical measurement procedure in Onyx.

this figure shows the general workflow of a physical measurement stage. For more details, see the section this figure.

General workflow of a physical measurement stage

Typical Steps in a Physical Measurement Stage

This section explains the steps in a typical physical measurement stage. It describes the steps shown in this figure in more detail.

1. **Contraindication page**. Optional. A study can check for contraindications if necessary.
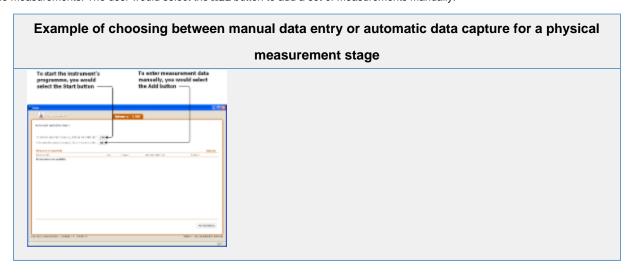2. **Instrument selection**. Only necessary if several instruments appropriate for the measurement are registered on the workstation. See Re gistering an instrument.
3. **Selection of automatic or manual measurement mode**. This is only necessary if both modes of capturing data are possible. For example, you may need to choose manual entry for an electronic instrument when the connection (between Onyx and the instrument) is not working.
4. **Automatic Data Capture**. For electronic instruments with their own software.
5. The user starts the instrument's programme from Onyx.
6. The instrument programme is displayed on the Onyx workstation.
7. The user takes the measurements using the instrument programme, and then closes the programme.
8. When the instrument programme closes, Onyx fetches the measurement data from the programme and displays it in Onyx.
9. **Manual Data Entry**. The user enters measurements taken manually. When the instrument is mechanical or when the connection to an electronic instrument with its own software is not working.
10. **Data Validation**. Optional. If the study defined data validation, Onyx validates the data, and if there are any problems, notifies the user so they can redo the measurement.
11. **Onscreen Report of Measurement Data**. The user can display a report of the measurement data (captured automatically or entered manually).
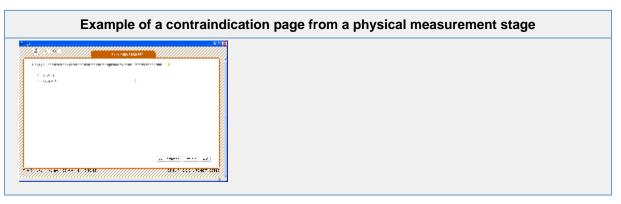
Example Physical Measurement Stage: Spirometry
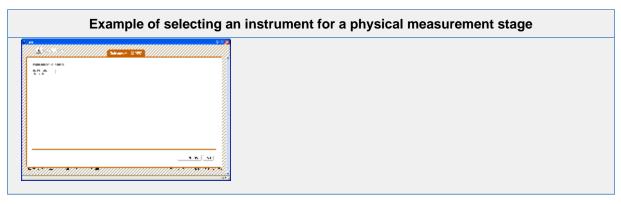
This section includes pages from an example physical measurement stage (a stage that collects spirometry measurements) and gives you a few pointers for recording physical measurements in Onyx.

The example `Spirometry` stage uses an instrument that is connected directly to the Onyx workstation. The instrument has its own programme that the user starts from Onyx. If the connection between the Onyx workstation and the spirometer is not functioning for some reason, the user could also read start the instrument's programme outside Onyx, take the measurements, and then enter them manually in Onyx.

1. **Contraindication page**. Optional. A study can check for contraindications if necessary. this figure shows an example contraindication page.

Example of a contraindication page from a physical measurement stage

2. **Instrument selection**. This is only necessary if several instruments appropriate for the measurement are registered on the workstation. If several instruments are available, you would have to select one of them by scanning or entering its barcode (and selecting enter) as shown in this figure.
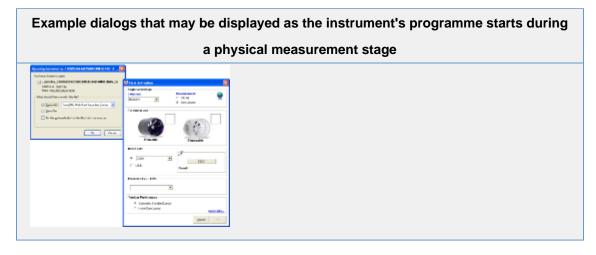

Example of selecting an instrument for a physical measurement stage

3. **Selection of automatic or manual measurement mode**. This is only necessary if both modes of capturing data are possible. For example, you would need to choose manual entry for an electronic instrument if the connection (between Onyx and the instrument) is not working. this figure shows an example of how a physical measurement stage could prompt the user to select automatic data capture or
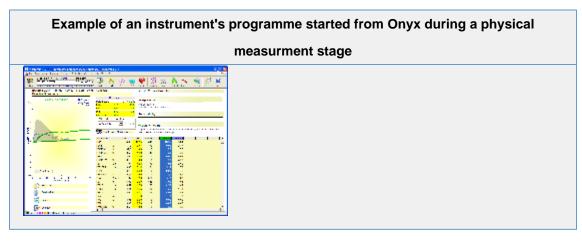
manual data entry. The user would select the `Start` button to start the instrument's own programme and then uses the programme to do the measurements. The user would select the `Add` button to add a set of measurements manually.
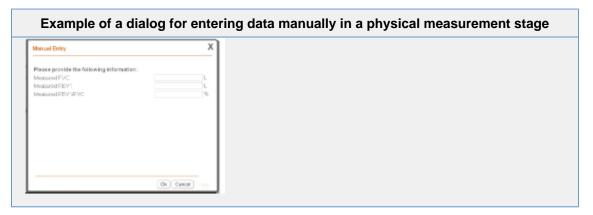
Example of choosing between manual data entry or automatic data capture for a physical
measurement stage

4. **Automatic Data Capture**. For a particular physical measurement stage, automatic data capture may be required or it may be an option that the user chooses. In either case, the user would have to start the instrument's programme and then the procedure would be similar to the following:
    1. After the user starts the instrument's programme from Onyx, it may take a moment or two for the programme to start. A couple of dialogs may be displayed while the programme is starting up. this figure shows some dialogs that might be displayed as the programme starts up. Usually, the user would select OK for such dialogs. The user would be trained how to use the instrument's software and could read the programme's documentation for more information.

Example dialogs that may be displayed as the instrument's programme starts during
a physical measurement stage
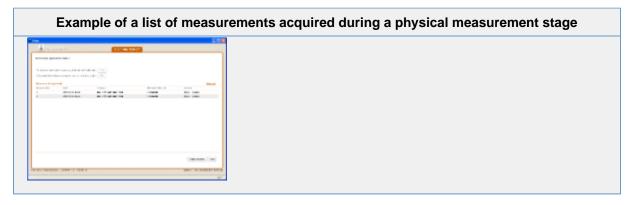
    2. The instrument programme is displayed on the Onyx workstation. this figure shows an example of an instrument's own programme. The user makes the necessary measurements using the instrument's programme.

Example of an instrument's programme started from Onyx during a physical
measurment stage
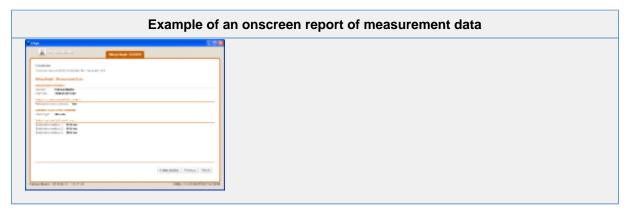
3. When the measurements are finished, the user would exit the programme.
4. When the instrument programme closes, Onyx fetches the measurement data from the programme.

5. **Manual Data Entry**. For a particular physical measurement stage, manual data entry may be required or it may be an option that the user chooses. Manual data entry is required for mechanical instruments and electronic instruments that do not connect directly to Onyx. It may be an option for electronic instruments that have their own programme for doing measurements. If the user enters the data manually, the procedure will be similar to the following:
    1. Onyx would display a `Manual Data Entry` page or dialog. this figure shows an example of such a dialog.

Example of a dialog for entering data manually in a physical measurement stage

---

    2. The user makes the necessary measurements, enters them in the dialog, and saves them.

6. **Validation**. If the study included data validation in the stage, Onyx would validate the data. If an error is detected, Onyx would display a message to inform the user about the error and to prompt them to make a new measurement.
   If a measurement passes validation, Onyx adds the measurement to a list displayed in Onyx. this figure shows an example of how measurements would be listed in Onyx.

Example of a list of measurements acquired during a physical measurement stage

---

7. **Onscreen Report of Measurement Data**. The last page of the stage includes a report of the measurement data (captured automatically or entered manually). this figure shows an example of how automatic measurements would be listed in Onyx.

Example of an onscreen report of measurement data

Additional Measurements. If additional measurements are required, the user would select `Start` again (to take another automatic measurement) or `Add` again (to enter another measurement manually).

Pointers for Physical Measurement Stages

Here are a good practices related to physical measurement stages:

- If your study has defined calibration procedures for any or all of the instruments used for physical measurement stages, your centre should calibrate the instruments following standard operating procedures and according to the schedule defined by your study. See Calibrating an instrument.
- Follow Onyx's on-screen instructions carefully.
- When working in an instrument's programme, follow on-screen instructions carefully and read the programme's user documentation for more information.
- If multiple measurements are required, be sure to acquire them all. Depending on how the stage has been customized, Onyx will probably prompt you if more measurements are needed.
- After you have obtained the required measurements, view the onscreen report and check that the data displayed in Onyx seems to be valid.

Collecting Biospecimens

Collecting Biospecimens

Biospecimens are often collected for research studies. In Onyx, biospecimens are collected during stages of the participant interview, so the general procedures for controlling an interview stage apply to biospecimen collection stages (see this figure).

> All Onyx interview stages, including those used to collect biospecimens, are highly customizable. Each study defines which biospecimens must be collected and establishes a way to determine if the collection of a particular biospecimen is contraindicated. Specimens that are routinely collected include blood, urine, and saliva. But your study could have defined collection of other types of biospecimen. Onyx allows studies to include directions to staff members right in the Onyx pages used to collect the biospecimens.

Due to the fact that each study's biospecimen collection stages are unique, this chapter cannot discuss the actual biospecimen collection stages defined by your study. Instead, this chapter includes:

- Key concepts that apply to all biospecimen collection stages
- An example of a biospecimen collection stage

Key Concepts

Multiple Samples

For a particular biospecimen collection stage, multiple samples of the particular biospecimen can be collected. For example, a study could require that five tubes of blood be collected.
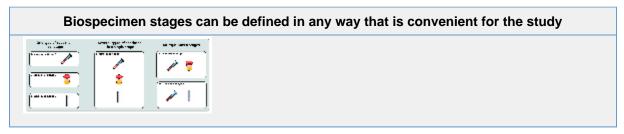
Multiple Stages

A study can define its biospecimen stages in any way that is convenient.

- A study can collect different types of biospecimen (blood, urine, saliva, and so on)
- A study can define one or more biospecimen stages
- Each stage can collect one or more types of specimen
- A particular type of specimen can be collected in one or more stages
- Several stages may be necessary to match contraindication constraints. For example, blood collection may be contraindicated but not urine collection, so it would be necessary to collect them in separate stages.
  this figure gives some examples of how biospecimen stages could be defined to suit a study's requirements.

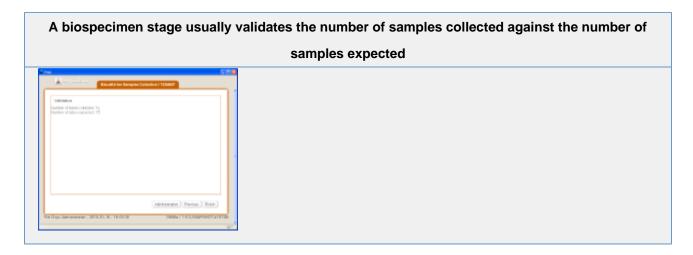**Biospecimen stages can be defined in any way that is convenient for the study**

☞ Sample Barcode

Typically, the receptacles used to collect biospecimens are labelled with barcodes that you scan during the biospecimen collection stage.

> Onyx can be customized to validate the format of the barcode. If the barcode does not pass the validation when you scan it, Onxy displays an error message.

☞ Samples Collected vs. Samples Expected

The last page of a biospecimen collection stage usually displays a report of the number of the number of *samples collected* (the number of barcodes that have been scanned) and the number of *samples expected* (the number of samples defined by the study). this figure shows an example page that reports this information. If too few samples were collected, you can select the `Previous` button to return to the page in which you can scan additional barcodes.

A biospecimen stage usually validates the number of samples collected against the number of samples expected

☞ Predefined Comments and Open Comments

Your study may have defined a number of comments that you can associate with a particular a sample. You choose the comment from a list. The use of predefined comments makes it possible to categorize the biospecimens collected.

Your study may also have provided a text box in which you can enter an open comment.

You will also be able to enter an open comment in the standard Onyx way when you finish the stage. You select `Adminstration` and then `Finish`. At that point an Action window will be displayed that contains a text field for any final comments you need to make about the stage. See Finishing a stage.
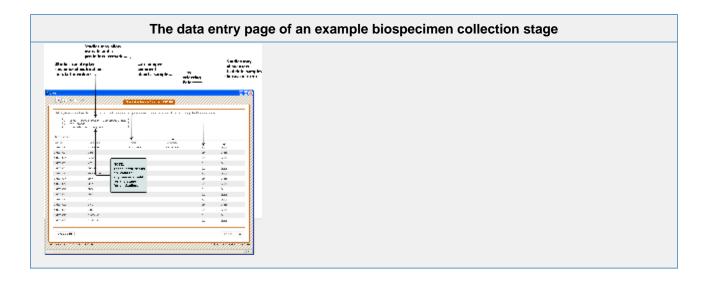
>_ Example Biospecimen Collection Stage

Apart from scanning the barcodes on sample receptacles, there is really nothing special about the procedures for biospecimen collection stages.

See this figure for the general procedures used to navigate through an interview stage.

Since your study has defined its own biospecimen collection stages, this guide cannot include procedures for those stages in your version of Onyx. Instead, this section uses an example blood and urine collection stage. this figure shows the data entry page of the example stage, and gives some pointers that apply to any biospecimen collection stage.

The data entry page of an example biospecimen collection stage

Managing Your Onyx User Profile

Managing Your Onyx User Profile

As an Onyx user, you can make a few changes to the way you interact with Onyx.
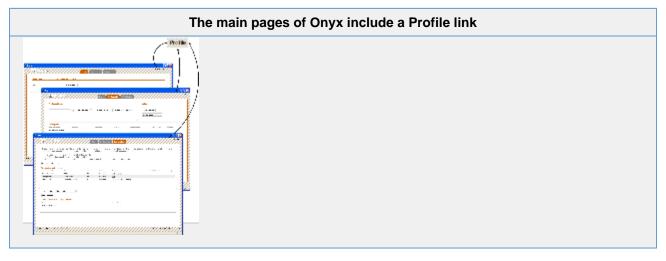
The information in this chapter apply to all Onyx users (data collectors, participant managers, and system administrators).
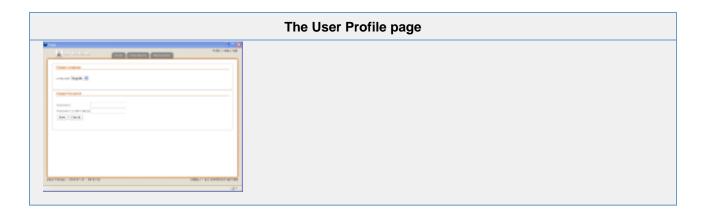
This chapter covers these topics:

- The user profile page
- When you cannot change your user profile
- Procedures:
  - Changing the language of the Onyx user interface
  - Changing your Onyx user password

The User Profile Page

The main pages of the Onyx user interface include a `Profile` link in their upper right corner.

The main pages of Onyx include a Profile link

When you select the `Profile` link, you see a page that allows you to view and change your user profile. this figure shows the user profile page.

The User Profile page

When You Cannot Change Your User Profile

The `Interview` page and the pages of interview stages do not include a `Profile` link which would give access to the user profile page. To ensure the integrity of interview data, Onyx verifies the interviewer's user name and password when most actions are taken during an interview. For this reason, it is not possible to change your user profile during the course of a participant interview. However, you can exit the interview, make changes to your profile, and then re-enter the interview. See the prerequisites in the Procedures section of this chapter.

Procedures

This section contains procedures related to managing your Onyx user profile.

Changing the Language of the User Interface

The pages of the Onyx user interface can be displayed in different languages. By default, several languages are currently available: English and French, Spanish, Catalan and Farsi.

> Your study may have translated the Onyx user interface or some parts of it (for example, a questionnaire) into additional languagesother than the default ones.

Prerequisite

If you need to change the language of the Onyx interface during the course of a participant interview, you must temporarily exit the interview as follows:

- If you're in the middle of an interview stage, pause the interview stage. See Pausing an interview stage.
- Exit the `Interview` page. See Exiting the Interview page.

After you change the language of the user interface, you can re-enter the participant's interview and resume the stage as explained in Entering the Interview page and Resuming a stage.

Procedure

1. Select the `Profile` link in the upper right corner of the Onyx page. Your user profile is displayed. If you do not see a `Profile` link, see the Prerequisite section.
2. Select a language from the drop-down list in the top half of the profile page. The language of the Onyx interface is changed immediately for the current page and for all other pages for which the selected language is available. If a label is not available in the selected language is not available, a default label is displayed.
3. Return to the rest of the Onyx user interface by selecting a tab (`Home`, `Particicpants` or `Workstation`).

Changing Your Password

if you know your current user password, you can change your Onyx user password as explained in this section.

If you do not remember your current user password, the system administrator can reset it to a temporary password, and afterwards you should follow this procedure to change it to a new password that only you will know.

Prerequisite

If you are in the middle of a participant interview, you cannot change your password. You must temporarily exit the interview as follows:

- If you cannot exit the interview, because you do not remember your current user password, see your system administrator.
- If you're in the middle of an interview stage, pause the interview stage. See Pausing an interview stage.
- Exit the `Interview` page. See Exiting the Interview page.

After you change your password, you can re-enter the participant's interview and resume the stage as explained in Entering the Interview page and Resuming a stage (that was paused).

Procedure

1. Select the `Profile` link in the upper right corner of the Onyx page. Your user profile is displayed. If you do not see a `Profile` link, see the Prerequisite section.
2. Enter the same new password in the `Password (confirmation)` field in the lower part of the page.
3. Select the `Save` button. Your new password is effective immediately.
4. Return to the rest of the Onyx user interface by selecting a tab (`Home`, `Participants` or `Workstation`).

Managing an Onyx Workstation

Managing an Onyx Workstation

An Onyx workstation is a key component in the experimental setup of your study. The `Workstation` page allows you to manage this important piece of research equipment.

The main reasons you need to manage a workstation are:

- *To set up the workstation so that participant interviews can be carried out as easily as possible.* For example, since a particular set of instruments is always used with a particular workstation, the `Workstation` page is used to register and reserve those instruments for the workstation. After instruments are registered and reserved for the workstation, Onyx will automatically identify the instrument that will be used for a measurement. Staff members do not have to scan an instrument barcode each time they take a measurement.
- *To store data about the experimental setup.* For example, to calibrate instruments and to log experimental conditions.

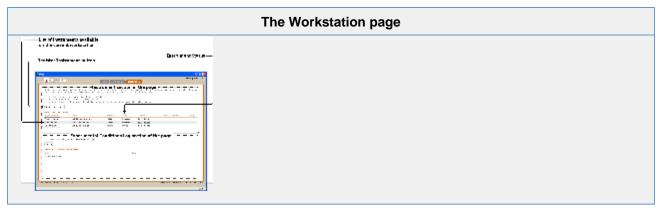This chapter covers these topics:

- The Workstation page and its important features
- Key Concepts related to the `Workstation` page
- Procedures related to instruments and experimental condition logs:
    - Registering an instrument
    - Setting the status of an instrument
    - Editing instrument information
    - Calibrating an instrument
    - Viewing the calibration log for an instrument
    - Exporting the calibration log of an instrument - available soon
    - Logging experimental conditions
    - Viewing experimental condition logs

The Workstation Page

The Onyx `Workstation` page allows you to register information about a particular Onyx workstationthe one on which you are currently working.

Mainly, you go to the `Workstation` page when you need to add or calibrate the instruments that are used to take physical measurements on the workstation and to log experimental conditions required for your study.

You access the `Workstation` page by selecting the `Workstation` tab. this figure shows the `Workstation` page and points out its main features.

The Workstation page

Here is an overview of the important features of the `Workstation` page:

- The **Instruments section** of the page allows you to register, edit, delete, and calibrate the instruments used with this workstation. This section contains:
    - The `Register Instrument` button opens a dialog that allows you to add an instrument to the list.
    - The `Instrument Status` indicates whether the instrument is available for the current workstation. The status can be: Reserved, Shared, or Out of service.

> The **Experimental Condition Log section** of the page allows you to record the experimental conditions that affect the instruments used to collect data. These logs are a customizable feature of Onyx. If your study has not defined any, the `Workstation` page will not have an Experimential Condition Log section.

Key Concepts

This section contains key concepts related to the `Workstation` page:

Instrument

A device used to take a physical measurement required by the study. All instruments that will be used with a particular Onyx workstation need to be registered on it. A particular instrument can be used for several types of measurement and during different physical measurement stages (for example, a tape measure). Even if an instrument is not electronic (for example, a grip strength dynamometer) or is not physically connected to the workstation (for example, an electronic scale), it must be registered on the workstation.

Laboratory instruments, used for the analysis of biospecimens, do not need to be registered on Onyx workstations.

Registering vs. Reserving an Instrument

Registering an instrument means adding it to the list of instruments used on the workstation on which you are currently working. See the procedure Registering an Instrument.

Reserving an instrument means that the instrument will only be used on one workstation. When you register an instrument on a particular workstation, its status is automatically set to Reserved, which means it will only be used with that workstation. If an instrument will be used with several workstations (for example, a grip strength dynamometer), its status must be set to `Shared` on one of the workstations. Then the instrument can be used with any workstation at the site. See Setting the Status of an Instrument.

Why Register Instruments?

The advantage of registering and reserving instruments (or setting their status to shared) is that Onyx will automatically identify the instrument used to take measurements during interviews. If no instrument is registered for a particular type of measurement, staff members will have to scan a barcode or enter an instrument ID manually before each measurement.

Instrument Status

The status of an instrument that has been registered on the current workstation meaning the workstation on which you are viewing the Workstation page.

An instrument's status appears in the `status` column of the `Workstation` page. The possible values for status are:

- **Reserved** This status means that the instrument is used exclusively on the current workstation. If an instrument's status is **Reserved**, the instrument will only be listed in the on the workstation on which it was registered.
- **Shared** This status means that the instrument can be used on the current workstation and other workstations at the site. If an instrument's status is **Shared**, the instrument will be listed in the Workstation page of all workstations at the site.
- **Out of service** This status means that the instrument is not available for some reason. The reason can be a technical problem or any other reason. If an instrument's status is **Out of Service**, the instrument will be listed in the Workstation page of all workstations at the site.

⚟ Calibration Log

A record of the dates and times at which an instrument was calibrated. The log contains an entry for each calibration that includes comments about whether the calibration was successful and if not, the error message that was generated.

> In order for it to be possible to calibrate an instrument, your study must have configured Onyx accordingly.

Procedures

This section contains procedures related to managing and calibrating instruments and maintaining logs of experimental conditions.

⚟ *Registering an Instrument*

When an instrument is going to be used with a particular workstation, you must register it on that workstation as explained in this section.

A particular instrument can be used for several types of measurement on a workstation. For example, a tape measure could be used for measurements in several different physical measurement stages. After you register an instrument for one measurement, you can register it for additional measurements as explained in Registering an Instrument for Additional Types of Measurement.

If an instrument is going to be used with several workstations, the instrument must be registered on each of those workstations as explained in this section. Then its status must be set to **Shared** on one of the workstations as explained in Setting the Status of an Instrument.
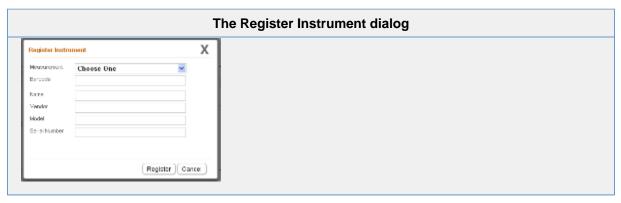
Prerequisites

To register an instrument, you need to know certain information about it. It is best to determine this information before you start the procedure:
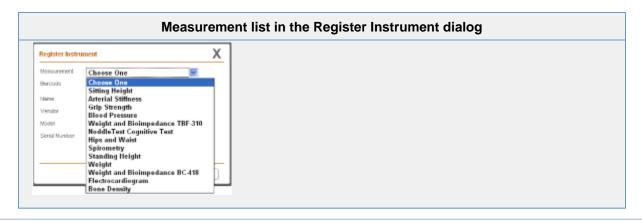
- **Measurement** The type of measurement that the instrument is used for. This information is required.
- **Barcode** You can use a scanner to complete this field or manually enter the number on a barcode label. This information is required. If the instrument does not have a barcode label, you must find a value to enter in this field.
- **Name** Your assessment centre may have come up with a scheme for naming instruments. It could include the instrument type and location. For example: Lab 1 Blood Pressure. This information is optional.
- **Vendor** The manufacturer of the instrument. This information is optional.
- **Model** Usually indicated on a label somewhere on the instrument. This information is optional.
- **Serial Number** Usually indicated on a label somewhere on the instrument. This information is optional.

Procedure

1. If the **Workstation** page (see this figure) is not displayed, select the **Workstation** tab to display it.
2. Select the **Register Instrument** button. The **Register Instrument** dialog is displayed. See this figure.
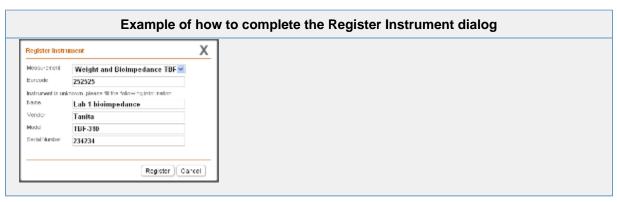

The Register Instrument dialog

3. Select the arrow beside the **Measurement** field and select the type of measurement for which the instrument is used.
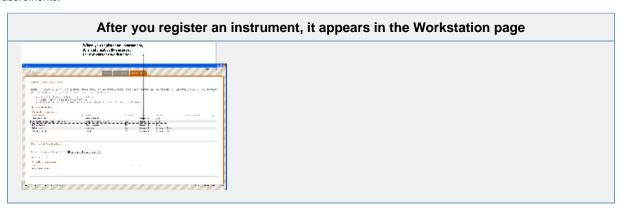
Measurement list in the Register Instrument dialog

The types of measurement that appear in the list is customizable so the list may contain different items in your version of Onyx.

4. Complete the `Barcode` field by scanning the barcode (the number is displayed in the field) or by entering a value manually (if you don't know what value to enter, see Prerequisites.
5. Use the mouse or tab key to select the `Name` field. When you do this, a message is usually displayed:*`Instrument is unknown, please enter the following information:`*You do not see the above message if the instrument was already registered on this workstation. Onyx automatically fills in the remaining fields on the page and you cannot edit themskip the next step.
6. Enter values in the `Name`, `Vendor`, `Model` and `Serial Number` fields. These fields are optional so you can complete them later if you don't have the information now. If you need help completing these fields, see Prerequisites. this figure shows an example.

Example of how to complete the Register Instrument dialog

7. If you decide not to register the instrument, select the `Cancel` button. The `Register Instrument` dialog closes and the `Workstation` page is visible again without any new items in the instrument list.
8. If you want to register the instrument, select the `Register` button. The `Register Instrument` dialog closes and the new instrument now appears in the list on the `Workstation` page. See this figureBy default, the instrument's status is set to Reserved. If the instrument will be used with several workstations, you must set its status to Shared as explained in Setting the Status of an Instrument.
9. If you need to register the instrument for another type of measure, do so as explained in Registering an Instrument for Additional Measurements.

After you register an instrument, it appears in the Workstation page

⚠️ *Registering an Instrument for Additional Measurements*
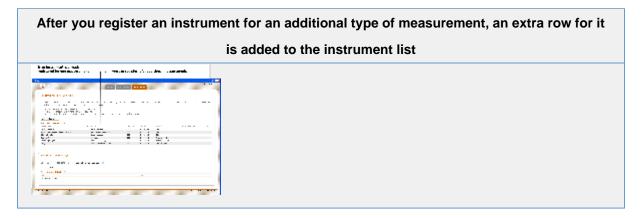
A particular instrument can be used for several types of measurement. For example, a tape measure could be used to measure waist and hips in one stage and to measure arm circumference in another stage. If an instrument has already been registered for one type of measurement, you can register it for additional measurements as explained in this section. When you scan or enter the barcode of the instrument, Onyx will recognize that it was already registered.

Procedure

1. If the `Workstation` page (see this figure) is not displayed, select the `Workstation` tab to display it.
2. Select the `Register Instrument` button. The `Register Instrument` dialog is displayed. See this figure.
3. Select the arrow beside the `Measurement` field and select the type of measurement for which the instrument is used. See this figure.
4. Complete the `Barcode` field by scanning the barcode on the instrument or by entering the value manually.
5. Use the mouse or tab key to select the `Name` field. When you do this, the remaining fields are filled in automatically with the values previously registered for the instrument. You cannot edit the values.
6. Check the information to confirm that it is the correct instrument.
7. If you decide not to register the instrument, select the `Cancel` button.
8. If you want to register the instrument, select the `Register` button. The `Register Instrument` dialog closes and a row for the new measurement appears in the list on the `Workstation` page. See this figure

After you register an instrument for an additional type of measurement, an extra row for it is added to the instrument list

Setting the Status of an Instrument

When an instrument is registered, its status is automatically set to Reserved for the workstation on which you registered it. You may need to change an instrument's status as explained in this section for one of these reasons:
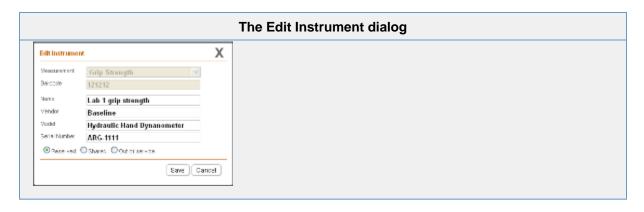
- If the instrument will be used with several workstations, you need to set its status to Shared on each of the workstations.
- If the instrument is defective or cannot be used for some reason, you need to set its status to Out of service.
- If an instrument was Out of service and is now ready to be put back in service, you need to set its status back to whatever it was before being taken out of service (either: Reserved or Shared).
- If a shared instrument will now be used with just one workstation, you only need to set its status to Reserved on that workstation (the instrument will automatically be removed from the other workstations with which it had been shared).
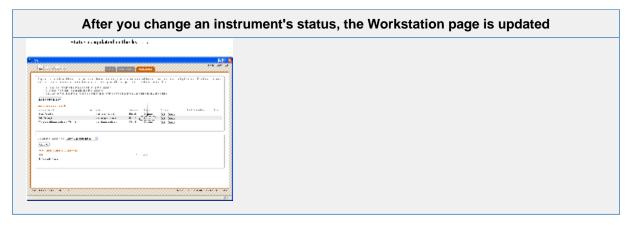
Prerequisites

To set an instrument's status on a particular workstation, the instrument must have been registered on that workstation, or its current status must be Shared or Out of Service.

Procedure

1. If the `Workstation` page (see this figure) is not displayed, select the `Workstation` tab to display it.
2. Select the `Edit` link for the instrument whose status you want to change. The `Edit Instrument` dialog is displayed.

The Edit Instrument dialog

3. Select the status you want to change to (`Reserved`, `Shared` or `Out of service`).
4. Click the `Save` button. The `Edit Instrument` dialog is closed and the `Workstation` page is redisplayed with the status changed to the value you just selected. See this figureIf you changed the status to `Shared` or `Out of Service`, the next time you look at the `Workstation` page on any of the other workstations on the site, that status will be displayed.

After you change an instrument's status, the Workstation page is updated

Editing Instrument Information

If you need to change any of the optional information that is stored for an instrument (name, vendor, model, serial number, status), you can do so as explained in this section.

You cannot change the barcode of an instrument after it has been registered. If the barcode is incorrect, you must delete the instrument (as explained in Deleting an Instrument) and register it again (as explained in Registering an Instrument).

Prerequisites

To change the information stored for an instrument for a particular workstation, the instrument must have been registered on that workstation, or its current status must be Shared or Out of Service.

Procedure

1. If the `Workstation` page (see this figure) is not displayed, select the `Workstation` tab to display it.
2. Select the `Edit` link for the instrument whose information you want to change. The `Edit Instrument` dialog is displayed. See this figure.
3. Change the values as necessary in any of the optional fields (`Name`, `Vendor`, `Model` and `Serial Number`). If you need help completing these fields, see Prerequisites.

Deleting a Measurement Type for an Instrument

If no measurements of a particular type were performed for an instrument, you can delete that measurement type for the instrument. The `Delete` link is only available if no measurements of that type were performed. See this figure. The study must keep information about any instrument that was actually used to acquire data. Deleting an instrument/measurement combination removes it from the list in `Workstation` page, as well as deleting its calibration log. The instrument itself is only deleted from the Onyx database when it is no longer associated with any measurement types.

You can delete a type of measurement for an instrument, if no measurements of that type have been made

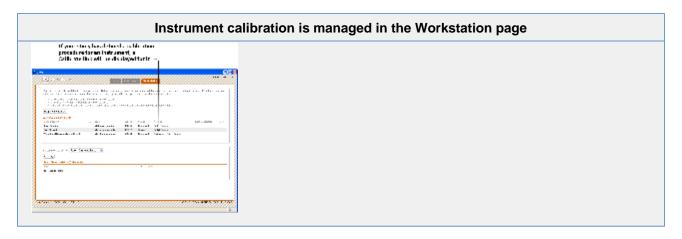### Procedure

1. If the `Workstation` page (see this figure) is not displayed, select the `Workstation` tab to display it.
2. Select the `Delete` link for the instrument/measurement that you need to delete. A dialog is displayed asking you to confirm that you want to delete.
3. Select `Yes` to confirm that you want to delete the instrument/measurement. The dialog closes and the `Workstation` page is visible againthe row for the instrument/measurement combination you just deleted has been removed from the instrument list.

Calibrating an Instrument

The instruments used for physical measurements may require calibration. Some instruments can be calibrated through their own software. Others must be calibrated manually. Onyx supports automated and manual calibrations, and allows each study to define custom calibration procedures for the instruments they use.

> Instrument calibration is a customizable feature of Onyx. If a calibration procedure has been defined for an instrument associated with a workstation, a `Calibrate` link will be available for that instrument in the `Workstation` page.

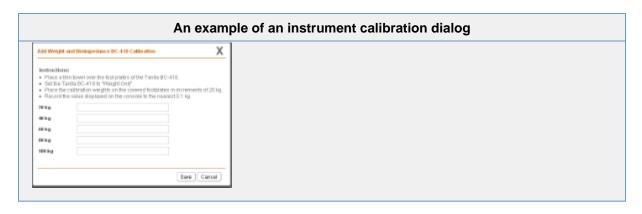**Instrument calibration is managed in the Workstation page**

Prerequisites

- Your study must have defined a calibration procedure for the instrument you want to calibrate.
- Since Onyx does not schedule instrument calibrations, you must know the instrument calibration schedule for your site.
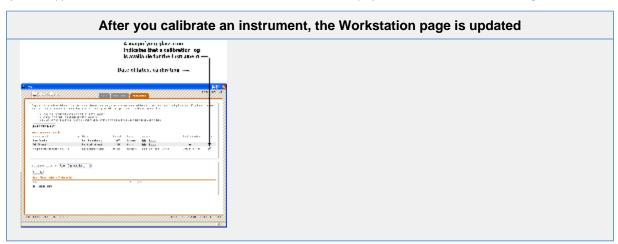
Procedure

1. If the `Workstation` page (see this figure) is not displayed, select the `Workstation` tab to display it.
2. Select the `Calibrate` link for the instrument that you want to calibrate. A calibration dialog is displayed. this figure shows an example of a calibration dialog.

An example of an instrument calibration dialog

3. Follow the directions in the calibration dialog box.
4. If you don't want to save the calibration data, select the `Cancel` button. The calibration dialog is closed and the `Workstation` page is visible again unchanged.
5. If you want to save the calibration data, select the `Save` button. The calibration dialog is closed and the `Workstation` page is visible. Today's date appears in the `Latest Calibration` column and an icon is displayed in the `Log` column. See this figure

After you calibrate an instrument, the Workstation page is updated

Viewing the Calibration Log for an Instrument
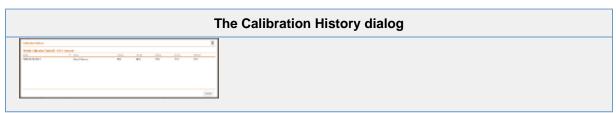
If an instrument can be calibrated, you can view a log of all the calibrations that have been done for it. A magnifying glass icon at the end of the row for an instrument in the `Workstation` page indicates that a calibration log is available for the instrument. See this figure.

Prerequisites

Your study must have defined a calibration procedure for the instrument you want to calibrate. At least one calibration must have been done.

Procedure

1. If the `Workstation` page (see this figure) is not displayed, select the `Workstation` tab to display it.
2. Select the icon (the magnifying glass icon) for the instrument whose log you want to view. The `Calibration History` dialog is displayed. See this figure.
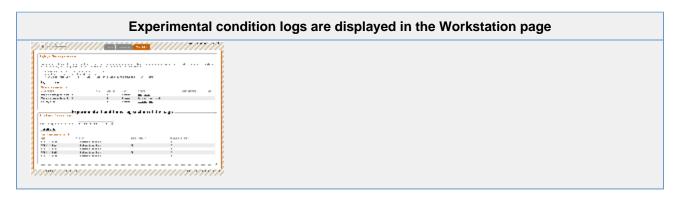

The Calibration History dialog

3. Select the `Close` button or the X button to close the dialog.

⚠ Logging Experimental Conditions

Onyx allows studies to log experimental conditions. These logs are used to capture any conditions that the study may wish to define, for example: room temperature and relative humidity.

Experimental condition logs are a customizable feature of Onyx. If your study has defined experimental conditions that it wants to log, the **Workstation** page will include an Experimental condition log section as shown in this figure. If the study has not defined any logs, the lower half of the **Workstation** page will be empty.

Experimental condition logs are displayed in the Workstation page
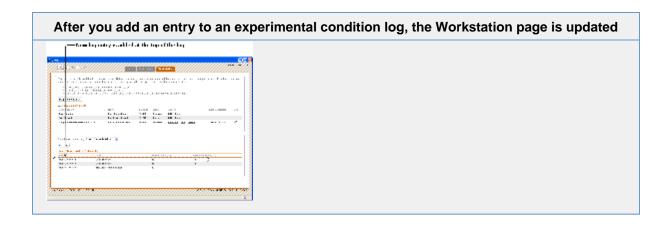

Prerequisites
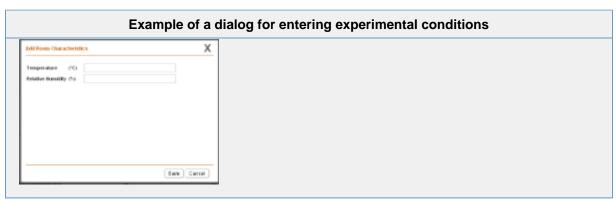
- Your study has defined at least one experimental condition log.
- Since Onyx does not schedule log entries, you must know when entries are supposed to be made for each experimental condition log defined for your study. For example: weather conditions should be logged every morning, and room conditions should be logged once a week, and so on.

Procedure

1. If the **Workstation** page (see this figure) is not displayed, select the **Workstation** tab to display it.
2. If your study has defined more than one log, select the log you want from the list labelled **Select a workstation log**. If the log you selected already contains some entries, they are displayed near the bottom of the **Workstation** page.
3. Select the **Add Log Entry** button. A customized dialog for entering log data is displayed. See the example in this figure.

Example of a dialog for entering experimental conditions

4. Complete the fields in the dialog box.
5. If you don't want to save the log entry, select the **Cancel** button. The dialog is closed and the **Workstation** page is visible again unchanged.
6. If you want to save the log entry, select the **Save** button. The dialog is closed and the **Workstation** page is visible. The new log entry appears at the top of the list of entries at the bottom of the page. See this figure.

After you add an entry to an experimental condition log, the Workstation page is updated

 Viewing Experimental Condition Log

Onyx allows studies to log experimental conditions. These logs can capture any conditions that a study may wish to define, for example: room temperature and relative humidity.
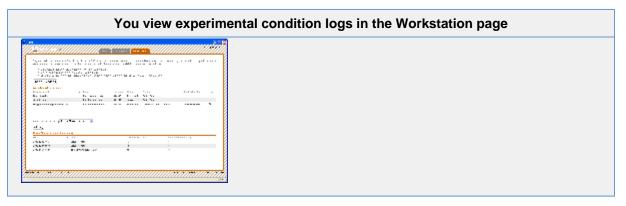
> Experimental condition logs are a customizable feature of Onyx. If your study has defined experimental conditions that it wants to log, the `Workstation` page will include an Experimental condition log section as shown in this figure. If the study has not defined any logs, the lower half of the `Workstation` page will be empty.

Prerequisites

Your study has defined at least one experimental condition log.

Procedure

1. If the `Workstation` page (see this figure) is not displayed, select the `Workstation` tab to display it.
2. If your study has defined more than one log, select the log you want from the list labelled `Select a workstation log`. The log is displayed at the bottom of the `Workstation` page. Log entries are displayed in reverse chronological order with the most recent entry at the top. See this figure

You view experimental condition logs in the Workstation page

Topics for Participant Managers

Topics for Participant Managers

This chapter covers topics of interest to Onyx users of type participant manager, including:

- Key concepts
- Procedures that can only be done by participant managers:
    - Receiving a participant
    - Enrolling a volunteer participant (customizable)
    - Editing a participant's registration information

- Updating the appointment list
- Unlocking an interview

Key Concepts

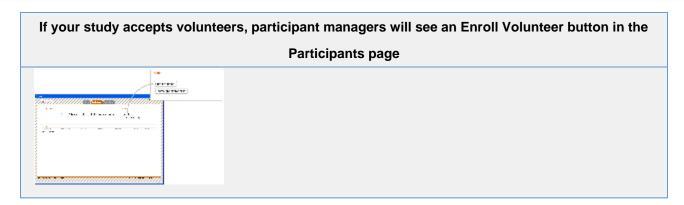This section contains concepts of interest to participant managers.

Receiving vs. Enrolling a Participant

Some studies invite people to participate in the study and set up appointments to interview them. These participants are added to the Onyx database when the appointment list is updated (see Updating the Appointment List). When a participant arrives at the centre for their interview, a participant manager receives them. *Receiving a participant* involves finding the participant in Onyx, assigning a Participant ID, and checking some information about the participant that Onyx already obtained from the appointment list. See Receiving a Participant.

Some studies accept volunteer or walk-in participants. When a volunteer participant arrives at an assessment centre, a participant manager must enroll them. *Enrolling a participant* involves assigning a Participant ID and entering some information about the participant. See Enrolling a Participant.

> Enrolling volunteer participants is a customizable feature of Onyx. If your study accepts volunteers, participant managers will see the `Enroll volunteer` button in the upper right corner of the `Participants` page. See this figure.

If your study accepts volunteers, participant managers will see an Enroll Volunteer button in the

Participants page

Procedures

Receiving a Participant

When a participant (who has been invited to participate in the study) arrives at the site for their interview, a participant manager must receive them as explained in this section.

You will have to assign a Participant ID number, the number that will be used to track the participant during the interview and, will probably be used to track their data for the duration of the study. It is important to review the list of participants carefully when registering a new participant and assigning them a Participant ID.
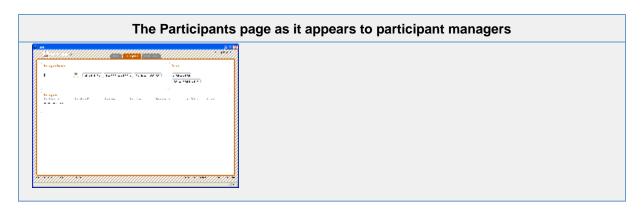
Prerequisites

- You must be an Onyx user of type participant manager
- The appointment list should have been updated recently to ensure that the participant is in the Onyx database. See Updating the Appointment List.
- You must be prepared to assign a new Participant ID to the participant.
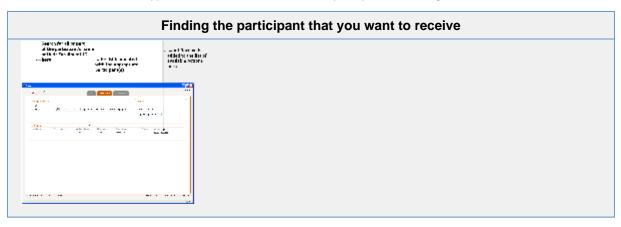
  > Each study determines how it will assign Participant IDs: either by scanning a barcode or by manually entering a number in a certain format and from a certain series of numbers.
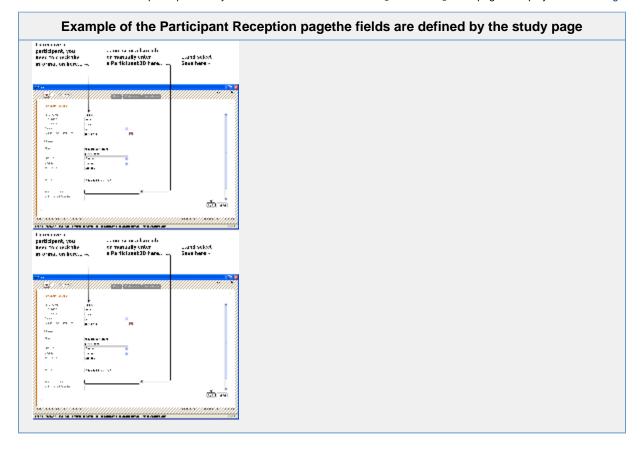
Procedure

1. If the `Participants` page is not displayed, select the `Participants` tab to display it. (see this figure)

The Participants page as it appears to participant managers

2. Search for the participant in any of the ways described in Finding a Participant. After you find the participant, their name is displayed in **P articipants** list and **Receive** appears in the **Actions** column for the participant. See this figure.

Finding the participant that you want to receive

3. Select the **Receive** link for the participant that you want to receive. The **Participant Reception** page is displayed. See this figure.

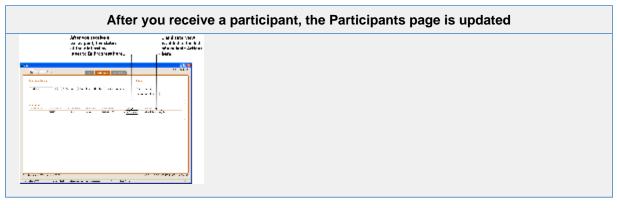## Example of the Participant Reception pagethe fields are defined by the study page

If Onyx has been configured to generate participant identifiers, the `Participant ID` field will have a `Generate ID` button next to it. Click on this button to automatically generate an ID for the participant being received. See this figure

Participant ID field with Generate ID button

4. If your study requires that you check a photo ID of the participant, now would be a convenient time to do so.
5. Check the information in the `Participant Registration` page with the participant and correct it, if necessary. Ensure that all required fields (marked with asterisks) are filled in. Fill in optional fields if the participant is willing to provide the information.

> The fields you see were defined by your study and may be different from those shown in this figure. Some fields are already completed with information that is in the appointment list. Certain key information (such as `Enrollment ID` in this figure) cannot be edited.

6. Assign a participant ID by scanning a new barcode or by entering a value manually in the `Participant ID` field.
7. If you decide not to proceed with receiving the participant, select the `Cancel` button. The `Participant Reception` page closes and the `Participants` page is redisplayed unchanged.
8. If you want to go ahead with receiving the participant, select the `Save` button. The `Participant Reception` page closes and the `Participants` page is redisplayed. The Status of the participant's interview has been updated to `In Progress` and `Interview` has been added in the `Actions` column for the participant. See this figure.

After you receive a participant, the Participants page is updated

9. If there are several pages of participants in the list, you can step through the pages in the list or search for the participant in order to confirm the change in their status or start the interview.

The participant can now be interviewed. See The Simplest Way to Start an Interview or Entering the Interview page.

Enrolling a Participant

> If your study accepts volunteer participants, a participant manager can enroll them as explained in this section.You will have to assign a Participant ID number, the number that will be used to track the participant during the interview and, will probably be used to track their data for the duration of the study. It is important to review the list of participants carefully when registering a new participant and assigning them a Participant ID.
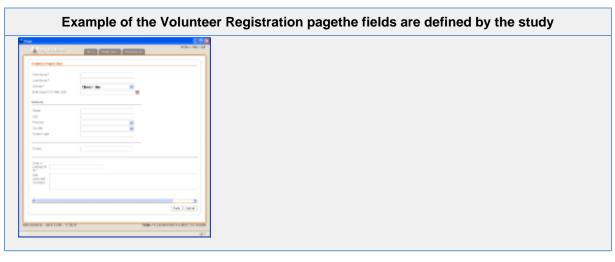
Prerequisites

- You must be an Onyx user of type participant manager
- Your study accepts volunteer participants. The `Enroll volunteer` button is displayed in the `Participants` page as shown in this figure.
- You must be prepared to assign a new Participant ID to the participant.

> Each study determines how it will assign Participant IDs: either by scanning a barcode, by manually entering a number in a certain format and from a certain series of numbers, or by automatically generating an ID.
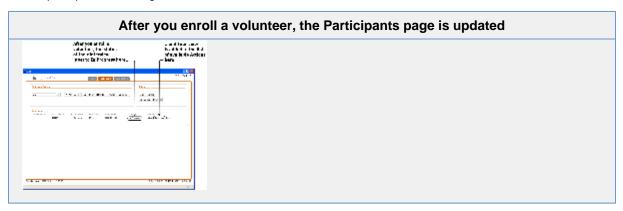
Procedure

1. If the `Participants` page is not displayed, select the `Participants` tab to display it. The `Enroll volunteer` button is displayed in the upper right corner.(see this figure).
2. Select the `Enroll volunteer` button. The `Volunteer Registration` page is displayed. See this figure.

> The fields you see were defined by your study and may differ from those shown in this figure.

Example of the Volunteer Registration pagethe fields are defined by the study

---

3. If your study requires that you check a photo ID of the participant, now would be a convenient time to do so.
4. Complete the fields. Required fields are indicated with an asterisk. You can scan a barcode to obtain a participant ID or enter a value manually.
5. If you decide not to enroll the participant, select the `Cancel` button. The `Volunteer Registration` page closes and the `Participants` page is redisplayed unchanged.
6. If you want to go ahead with enrolling the participant, select the `Save` button. The `Volunteer Registration` page closes and the `Participants` page is redisplayed. The participant's status updated to `In Progress` and `Interview` has been added in the `Actions` column for the participant. See this figure.

After you enroll a volunteer, the Participants page is updated

---

 Enrolling a Participant from an External System

If Onyx has been configured to look up participant information stored in an external system (a "participant registry"), the enrollment procedure is slightly different.
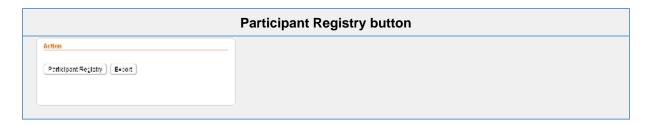
You will first search for the participant in that system, using whatever unique identifier that system requires. After that, you will assign to that participant a Participant ID.

Prerequisites

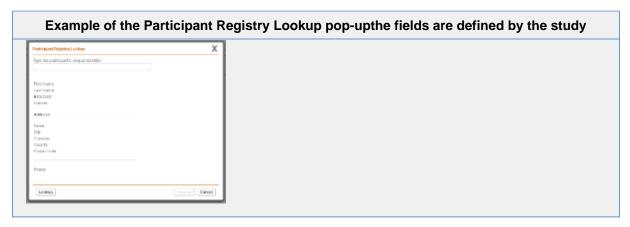The prerequisites are the same as in "regular" enrollment.

Procedure

1. The `Participant Registry` button is displayed in the upper right corner of the `Participants` page. See this figure.

Participant Registry button

2. Select the `Participant Registry` button. The `Participant Registry Lookup` pop-up window is displayed. See this figure.

> The fields you see were defined by your study and may differ from those shown in this figure.

Example of the Participant Registry Lookup pop-upthe fields are defined by the study

3. Enter the participant's unique identifier and select `Lookup`. Upon a successful lookup, the participant fields will be filled in with the retrieved information.
4. To enroll the participant, select `Receive`. The `Volunteer Registration` page will be displayed. Now proceed the same way as with "regular" enrollment.
5. If you decide not to enroll the participant, select the `Cancel` button. The `Participant Registry Lookup` pop-up window closes and the `Participants` page is redisplayed unchanged.

Editing a Participant's Registration Information

After a participant has been received or enrolled, a participant manager can edit their registration information. It is potentially dangerous to change a participant's registration information after their interview has been started. For example, changing the participant's gender could cause a problem since some physical measurements and questionnaires depend on it. Onyx does not prevent you from creating this kind of problem.

> Each study defines the registration information that is stored for a participant. Certain key information (such as the participant's name and ID numbers) cannot be changed after the participant was received or enrolled. Other information can be modified.

Prerequisites

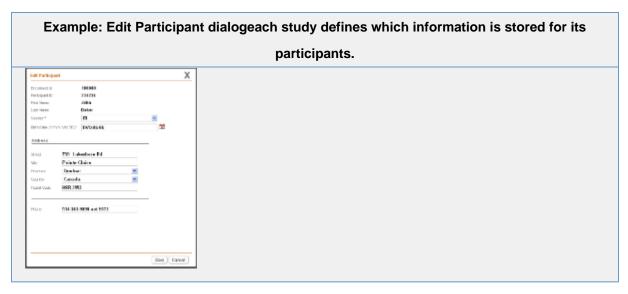- You must be an Onyx user of type participant manager

Procedure

1. If the `Participants` page is not displayed, click the `Participants` tab.
2. Find the participant whose information you want to edit. If you need help finding the participant, see Finding a Participant. When the `Participants` list is displayed, an `Edit` link appears (in the `Actions` column) for the participant. See this figure.

3. Select the `Edit` link for the participant. The `Edit Participant` dialog is displayed. See this figure.

> Each study defines the information that is stored for its participants. You may not see the same fields as those shown in this figure.

Example: Edit Participant dialogeach study defines which information is stored for its participants.

4. Make the necessary changes. Certain key information (such as the participant's name and ID numbers) cannot be edited.
5. If you decide not to store the changes, select the `Cancel` button.
6. If you want to proceed with changing the information, select the `Save` button. The `Edit Participant` dialog closes. Your changes have been stored in the Onyx database.

Updating the Appointment List

Onyx does not schedule appointments for participants. Instead it imports an appointment list from a file. The appointment list is updated automatically. If necessary, Onyx users of type participant manager can also update the appointment list manually as explained in this section.

> Your study customized how often the appointment list is updated. Contact your system administrator if you want to know how often the list is updated or have any other questions about the list.
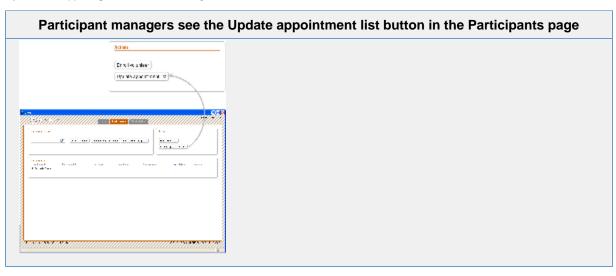
Prerequisite

- You are an Onyx user of type participant manager.
- If you want to select an appointment list file, you must know the name and location of such files. Appointment files usually have an XML or XLSL extension.

Each study customizes its appointment list. Your study defined the file type, name, and location, as well as the information stored for each participant. Contact your system administrator to know more about appointment list files for your study.

Procedure

1. If the `Participants` page is not displayed, select the `Participants` tab to display it. The `Update appointment list` button is displayed in the upper right corner. See this figure.
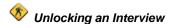
   Participant managers see the Update appointment list button in the Participants page

2. Select the `Update appointment list` button. The `Update Appointments` dialog is displayed. See this figure.

   The Update Appointments dialog

3. If you want to use the appointment file from the default location, select the first option button (`Update appointments with the latest appointment file available`).
4. If you want to search for an appointment list file to use for the update, select the second option button (`Update appointments with an alternative appointment file`). If you need help locating a file, contact your system administrator.
5. If you decide not to update the list, select the `Cancel` button. The `Update Appointments` dialog is closed and the `Participants` page is visible again.
6. If you want to proceed with updating the list, select the `Update Appointments` button. During the update, a progress bar is displayed in the dialog. When the update is complete, the dialog displays a report about the update. See this figure.

   The Update Appointments dialog displays a report when the update is complete

7. Select the `Details` button if you want to read more about how the update went. A `Log` dialog is displayed. If any errors were reported in the log, contact your system administrator. When you have finished viewing the log, select the `Close` button of the dialog.
8. When you have finished viewing the report in the `Update Appointments` dialog, select its `Close` button. The dialog is closed and the `Participants` page is visible again. The list of participants is now up-to-date.
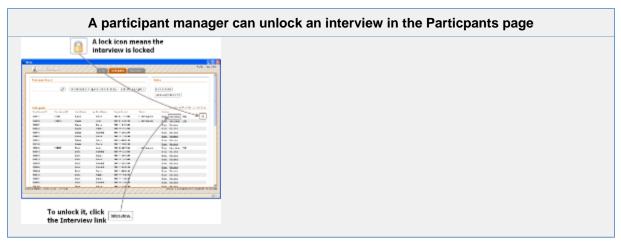
Unlocking an Interview

When a staff member is in the middle of an interview stage with a participant, that staff member has a lock on the interview. This prevents conflicts in storing data. Occasionally, you may need to unlock an interview. For example, a technical problem may be preventing the interview from being released, or the staff member who has a lock on the interview is not able to continue the interview. A participant manager can unlock the interview as explained in this section.

Prerequisite

You are an Onyx user of type participant manager.

Procedure

1. If the `Participants` page is not displayed, select the `Participants` tab to display it.
2. Locate the participant whose interview is locked. See this figure.

A participant manager can unlock an interview in the Particpants page

3. Select the `Interview` link for the participant whose interview you want to unlock. A message is displayed asking you to confirm that you want to unlock the interview.

A message asks you to confirm that you want to unlock the interview

4. If you decide not to unlock the interview, select the `No` or `Cancel` button. The `Participants` page is visible displayed. The interview is still locked.
5. To unlock the interview, select the `Yes` button. The `Interview` page for the participant is displayed.
6. To allow another Onyx user to work on the interview, select the `Exit X` button. The `Interview` page closes. The `Participants` page is visible displayed. The interview is unlocked.

Topics for System Administrators

Topics for System Administrators

This chapter covers topics of interest to Onyx whose role is system administrator. A system administrator has all the permissions of the other two user roles (data collector and participant manager), as well as a few additional permissions. This chapter covers those additional permissions. In particular, this chapter covers:

- Key concepts of interest to system administrators
- Procedures (in the Onyx user interface) that can only be done by system administrators:
    - Adding an Onyx user
    - Editing the profile of an Onyx user (including password reset)
    - Activating and deactivating Onyx users

- Deleting an Onyx user
- Exporting data (participant data, as well as data about instruments and experimental conditions)
- Purging data (participant data only)

Key Concepts

This section contains concepts of interest to system administrators.

User Roles

These are the roles that are possible for Onyx users:

- Data collector - has basic permissions
- Participant manager - has data collector permissions plus a few more
- Questionnaire editor - only has permissions to create and edit questionnaires
- System administrator - has permissions of participant manager plus a few more

For details about what each user role can do, see Who Uses Onyx.

User Status

Onyx users have one of these statuses:

- **Active**: The user can log in. By default, when a user is created, their status is active.
- **Inactive**: The user will not be able to log in. For example, you may want to deactivate a user who is away on leave.

The word that appears in the `status` column in the `User` administration page is the user's current status.

Only system administrators can change the status of an Onyx user. See Activating and Deactivating Onyx Users.

Data Export

Exporting data from Onyx means reading data from the Onyx database and writing it to one or more export destinations. Here are some points worth noting about the Onyx export function:

- Exporting does not delete any data from the Onyx database. Deleting data from the database is done by the purge function. See Data Purge.
- An export destination is a compressed zip file. The name given to an export destination may indicate the location into which the data will be importedfor example, a data repository like Opal.
- Participant data and experimental conditions data can be exported. See Participant Data and Experimental Condition Data.
- Configuration of data export is done entirely in configuration files, not through the Onyx user interface. Some things that can be configured:
    - Which data is selected for export
    - Directory to which export files are written
    - How many export destinations are defined
    For detailed information about how the export function is configured, see the OBiBa Wiki (http://wiki.obiba.org/confluence/display/ONYX/Onyx+1.6.0+Upgrade).

Only system administrators can execute a data export from the Onyx user interface. See Exporting Data.

Data Purge

In Onyx, purging data means deleting data from the Onyx database. Only participant data can be purgednot experimental conditions data.

Configuration of data purging is done entirely in configuration files, not through the Onyx user interface. For detailed information about how the purge function is configured, see the OBiBa Wiki (http://wiki.obiba.org/confluence/display/ONYX/Onyx+1.6.0+Upgrade).

Only system administrators can execute a purge from the Onyx user interface. See Purging Data.

Participant Data

Participant data includes personal information (such as the participant's name and address), as well the data from all stages of the interview (consent, questionnaires, physical measurements, and information about the biospecimens collected).

Participant data can be exported (see Exporting Data) and purged (see Purging Data).

Experimental Conditions Data

Each study can define experimental conditions that it would like to track on a regular basis. This information is stored in one or more experimental condition logs. The `Workstation` page of the Onyx user interface allows users to make entries in these logs. As a system administrator, you will export this data, so you should be familiar with the experimental condition logs that were defined for the study. The user interface for experimental condition logs is in the lower half of the `Workstation` page.

Instrument data is a special case of experimental conditions data. See Instrument Data.

Experimental conditions data can be exported (see Data Export), but it cannot be purged (see Data Purge). This data is not purged because it is needed for the log history, and because no privacy issues are involved with this kind of data.

Instrument Data

Instruments are part of the experimental setup, so data about instruments is a special case of experimental conditions data. As a system administrator, you will export this instrument data, so you should be familiar with the instrument calibrations that are done for the study. The user interface for instrument calibration is in the the top half of the `Workstation` page. Like other experimental condition data, instrument data can be exported but not purged.

Procedures

This section contains procedures that require system administrator permissions.

Adding an Onyx User

Only system administrators can create new Onyx users. You must know what role the user will play: Data collector, Participant Manager or System Administrator. For details about what each user role can do, see Who Uses Onyx.

Procedure

1. Log in to the Onyx user interface.
2. Select the `Administration` link in the upper right corner of any Onyx page. By default, you are on the `User` page.
3. Select the `Add User` button. A dialog prompts you to enter information about the new user.
4. Enter the appropriate information in the user identification fields.
5. In the `Role(s)` list, select all the roles that the user will play. To select multiple roles, hold down the `Ctrl` key and click on each of the roles.
6. For `Language`, select the language that the user prefers for viewing the Onyx user interface.
7. Click*`Save`*.

Editing the Profile of an Onyx User

Only system administrators can edit the profile of an Onyx user as explained in this section. You can reset the user's password when you edit their profile.

Procedure

1. Log in to the Onyx user interface.
2. Select the `Administration` link in the upper right corner of any Onyx page. By default, you are on the `User` page.
3. Select the `Edit` link for the user whose profile you need to edit. A dialog displays the information you can edit.
4. Make the necessary changes and select `Save`.

Activating and Deactivating Onyx Users

You may need to deactivate a user temporarily. You do this by changing their status to Inactive. When you need to reactivate a user who was deactivated, you change their status to Active. The word that appears in the `status` column in the `User` administration page, is the user's current status.

If you need to remove a user permanently, you must delete them. See Deleting an Onyx User.

Procedure

1. Log in to the Onyx user interface.
2. Select the `Administration` link in the upper right corner of any Onyx page. By default, you are on the `User` page.
3. Locate the user whose status you need to change. The word that appears in the `status` column is the user's current status.
4. Click on the user's current status in the Status column. The status changes immediately to the new status. If the status was `Active`, it

changes to **Inactive**. If the status was **Inactive**, it changes to **Active**.

⚠ Deleting an Onyx User

Deleting an Onyx user removes them from the user list permanently. Only system administrators can delete an Onyx user as explained in this section.

You can also deactivate a user temporarily, rather deleting them, if this is more appropriate. See Inactive user status, and Activating and Deactivating Onyx Users.

Procedure

1. Log in to the Onyx user interface.
2. Select the **Administration** link in the upper right corner of any Onyx page. By default, you are on the **User** page.
3. Select the **Delete** link for the user that you need to delete. A dialog prompts you to confirm that you want to delete the user.
4. Select **OK** to delete the user.

⚠ Exporting Data

Exporting data from Onyx means copying participant and experimental condition data (including instrument data) from the Onyx database to one or more export destinations. Only system administrators can execute an export as explained in this section.

For information about what is exported when you click the **Export** button, see Data Export.

Procedure

1. Log in to the Onyx user interface.
2. Select the **Administration** link in the upper right corner of any Onyx page.
3. Select the **Data** tab.
4. Select the **Export** button. A dialog prompts you to confirm that you wish to export.
5. Select the **Yes** button . A dialog prompts to confirm that you wish to proceed with the export.

⚠ Purging Data

In Onyx, purging data means deleting data from the Onyx database. Only system administrators can execute a purge as explained in this section.

For information about what is deleted when you click the **Purge** button, see Data Purge.

Procedure

1. Log in to the Onyx user interface.
2. Select the **Administration** link in the upper right corner of any Onyx page.
3. Select the **Data** tab.
4. Select the **Purge** button. A dialog prompts you to confirm that you wish to purge participant data.
5. Select the **Purge Participants** button. A dialog prompts you to confirm that you wish to proceed with the export.

Onyx Developer Guide

Contents of this Guide

- Introduction
- Contributing to Onyx Source Code
- Instrument Integration
- Participant Registry

Introduction

This guide is for whoever will contribute to Onyx development.

Contributing to Onyx Source Code

The base code of Onyx source is on GitHub: https://github.com/obiba/onyx.

The instrument integrations examples can be found in Onyx Instruments repository on GitHub: https://github.com/obiba/onyx-instruments

Contributors to Onyx must follow the Fork-Pull-request pattern of GitHub. More information about this development model is available on GitHub:

- Fork A Repo
- Using Pull Requests

Development environment:

- Onyx uses Maven as Project management tool. Maven version 3+ is required.
- Java SDK 1.8+ is required.
- Recommended IDE is IntelliJ or Eclipse

Compiling Onyx:

```
mvn clean install
```

Testing Onyx:

```
cd onyx-integration-test
mvn jetty:run-war


# then connect to http://localhost:8080
```

Instrument Integration

Integrating an instrument in Onyx requires some development:

- Onyx project provides an instrument integration framework,
- The Instrument Integration Project to be developed will define:
    - The executable for instrument data extraction (Java and any other language that would be appropriate for this instrument),
    - The integration of the instrument in Onyx server (XML).

Participant Registry

Documentation is in progress

Integrating an Instrument

Contents of this Guide

Introduction

This guide will help developers in their process of integrating an instrument in Onyx workflow:

- from an interview instrument stage, start an electronic instrument,
- extract the data and send them to the Onyx server.

Developer is expected to be familiar with Java ecosystem (especially Maven) and Git (especially GitHub).

Framework

Onyx provides a complete framework for integrating instruments to Onyx workflow.

Architecture

The set-up is the following:

- Onyx server: the web server which will authenticate operator, will control the interview workflow, will serve the InstrumentRunner application binaries and will retrieve/store the participant's data,
- Workstation: the operator's computer from which the participant is interviewed. At least two applications runs on this machine:
    - a web browser (Chrome, Firefox) that connects to Onyx web server,
    - a InstrumentRunner application that is launched from the web browser using Java Web Start and which is responsible for handling the communication between the instrument and the Onyx server,
    - the instrument application provided by the vendor (if any).
- Instrument: the instrument is physically connected to the workstation (can be usb, ethernet, serial etc.) and all communication with the Onyx server will go through this workstation.

Instrument

An instrument is composed of:

- a device (always): Onyx is able to integrate only electronic devices
- a software application (sometimes): provided by the vendor and which controls the device. The instrument software application is optional: it is not required for Onyx integration, but if it exists one should try to extract data from this application instead of accessing directly the device (because the application usually performs some adjustments/corrections/calculations on the data extracted from the device). Due to lack of communication capabilities (proprietary formats, legal issues, no API) of some instrument applications, it is not always possible to control it.
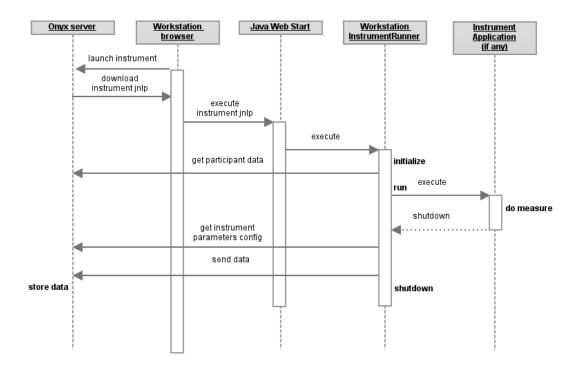
Java Web Start

The InstrumentRunner application is launched from the operator web browser using the Java Web Start (JWS) service. Onyx server provides the `jnlp` file that describes to JWS the application to be launched (executable, libraries) and what are the security settings for this application.

The `jnlp` file served by Onyx is generated from a template defined in onyx-instruments project. This template can be overridden on a per instrument integration basis (see GE Vivid as an example).

Instrument Execution Context

Spring is used for handling the HTTP communication between the InstrumentRunner application and the Onyx web server in the context of the operator web session.

Sequence Diagram

Java API

Spring handles the following beans:

InstrumentRunner

Each instrument integration project must provide an implementation of the InstrumentRunner interface:

- `initialize`: prepare the instrument (open communication channel, push participant data, backup database and configuration files etc.)
- `run`: do the measure, extract, prepare and send data to the server.
- `shutdown`: shutdown the instrument (close communication channel, restore database and configuration files etc.)

InstrumentExecutionService

The InstrumentExecutionService is the service to be used for communicating with the Onyx server, i.e. for:

- getting instrument configuration,
- getting participant's data,
- getting operator's data,
- sending measure's data.

ClientLoggingService

The ClientLoggingService is the service to be used for logging messages both on client and server side.

ExternalAppLauncherHelper

The ExternalAppLauncherHelper is a helper class for launching the instrument software application (if any).

Instrument Integration Project

Your instrument integration project will be a sub-project of `onyx-instruments` project:

https://github.com/obiba/onyx-instruments

Design

Each instrument is specific. To help you to know how to integrate it with Onyx you will have to answer to the following questions:

- Is software provided with the instrument and how to execute it?
- How to know when the measure is finished?
- Is it required by the instrument to have some participant's data (age, gender etc.) before starting the measure and how to push those data?
- How to extract data from the instrument (serial port, network, database, file etc.)?
- What is the format of the data (xml, proprietary binary format etc.)?
- How to remove participant's data when measure is finished?
- Which instrument version is to be integrated?
- Which Operating System will run on the operator workstation?
- What is the associated SOP?
- Which data are to be collected and what is their type (integer, decimal, text, binary, boolean)?
- How many measures must be done during one stage?

Implementation

Project Structure

Suggested Maven project structure is the following:

```
interface-<measure>-<instrument>/
 pom.xml
 src
     main
         java
             org
                 obiba
                     onyx
                         jade
                             instrument
                                 <measure>
                                     MyInstrumentRunner.java
                                     MyParser.java
                                     ...
         resources
             instrument_en.properties
             instrument_fr.properties
             META-INF
                 onyx
                     defaults.properties
                 spring
                     instrument-context.xml
     test
         java
             org
                 obiba
                     onyx
                         jade
                             instrument
                                 <measure>
                                     MyParserTest.java
         resources
             test_file1
             test_file2
             ...
```

Where:

File	Description
`MyInstrumentRunner.java`	Required [InstrumentRunner](#) implementation for the instrument.
`instrument_*.properties`	Localization strings when [InstrumentRunner](#) application provides a GUI.
`default.properties`	If instrument has properties (like some configuration or executable file location, database driver type etc.) that can be overridden by the Onyx server that will embed this instrument integration, the default values of these properties can be declared in that file and referenced by their key in the `instrument-context.xml`. See for instance [Weight-RiceLake default.properties](#).

`instrument-context.xml`	Required Spring beans declaration file. Does the wiring between your InstrumentRunner implementation and the InstrumentExecutionService (plus the locale of the operator). Other properties can also be set in the #InstrumentRunner bean (default values being declared in the `default.properties`) file. See for instance Weight - RiceLake instrument-context.xml.
`MyParser.java`	If relevant, data extraction class that parsers some string received from the instrument.
`MyParserTest.java`	Unit testing of the parser, using test files.

Unit Tests

As much as possible and especially when some proprietary format are involved, add some unit testing classes to validate data extraction quality and prevent regression when upgrading instrument (software or hardware) version.

Integration Tests

You have to set-up a specific project for building a Onyx web application that will embed your instrument application: onyx-integration-test project should be used for this. This project provides an example of the Onyx configuration required to use your instrument integration.

The project structure looks like:

```
onyx-integration-test/src/main/webapp/WEB-INF/config/jade/
  resources
      instruments
          <measure>
              instrument-descriptor.xml
              messages_en.properties
              messages_fr.properties
          ...
  stages.xml
```

Where:

File	Description
`instrument-descriptor.xml`	Declares the instrument parameters (input and output), validations, multiple measures etc.
`stages.xml`	Declares one stage per instrument.

The integration project `pom.xml` is also to be updated as in the onyx-integration-test example:

```
<dependencies>
    ...
    <dependency>
      <groupId>org.obiba.jade.instrument</groupId>
      <artifactId>interface-bonedensity-holologic-apex</artifactId>
      <version>${project.version}</version>
      <type>zip</type>
      <scope>provided</scope>
    </dependency>
    ...
  </dependencies>

  <build>
    ...
    <plugins>
      <plugin>
        <artifactId>maven-dependency-plugin</artifactId>
        <executions>
          <execution>
            <id>unpack-instruments</id>
            <phase>process-resources</phase>
            <goals>
              <goal>unpack</goal>
            </goals>
            <configuration>
              <artifactItems>
                ...
                <artifactItem>
                  <groupId>org.obiba.jade.instrument</groupId>

<artifactId>interface-bonedensity-holologic-apex</artifactId>
                  <type>zip</type>
                  <outputDirectory>

${project.build.directory}/${project.build.finalName}/instruments/BoneDens
ityAPEX

                  </outputDirectory>
                </artifactItem>
                ...
    </plugins>
  </build>
```

See also documentation about Configuring Physical Measurement Stages.

Once the integration test project is set-up, run the following command to start a Onyx web server in development mode:

```
cd onyx-integration-test
mvn jetty:run-war
```

Then connect to http://localhost:8080 (or http://:8080 if operator workstation runs a different OS (usually instrument applications are available on Windows only)) and test (receive, interview dummy participant, start instrument stage and launch instrument runner application through JWS).

Examples

Measure	Instrument	Communication	Data format	User Interface	Comment
Ankle Brachial	Vantage ABI	File	Proprietary Format (hexadecimal)	GUI provided by Onyx	Output file is selected manually by the operator
Arterial Stiffness	Sphygmocor	MS Access Database		Instrument application	
Audiogram	Tremetrics RA300	Serial Port	Proprietary Protocol	GUI provided by Onyx	Uses RXTX library for serial port communication
Bioimpedance	Tanita TBF310	Serial Port	Proprietary Protocol	GUI provided by Onyx	Uses RXTX library for serial port communication
Bioimpedance	Tanita BC418	Serial Port	Proprietary Protocol	GUI provided by Onyx	Re-use of some code defined for TBF310
Blood Pressure	VSM BPTRU	USB	Proprietary Protocol	GUI provided by Onyx	Uses HID API for USB port communication
Bone Density	Achilles Express	MS Access Database		Instrument application	
Bone Density	Hologic APEX	MS Access Database, DICOM Server		Instrument application	Uses dcm4che for DICOM processing
Cognitive Test	NoddleTest	File	Proprietary Format (text)	Instrument application	No device
ECG	CardioSoft	File	XML	Instrument application	
ECG	CardioSoft MAC800	File	XML	Instrument application	
Grip Strength	JTech	Paradox Database		Instrument application	
Retinal	Topcon Imagenetr4lite	MS SQLServer Database		Instrument application	
Spirometry	Minispir	File	Proprietary Format (text)	Instrument application	
Spirometry	NDD EasyOnPC	File	XML	Instrument application	
Tonometer	Reichert ORA	MS Access Database		Instrument application	
Ultrasound	GE Vivid	DICOM server		GUI provided by Onyx	Uses dcm4che for DICOM processing. Default JNLP template overridden to increase heap size.
Weight	RiceLake 140-10	Serial Port	Proprietary Protocol	GUI provided by Onyx	Uses RXTX library for serial port communication