Rave Architect “Lite”

Instructor: [Name]

About this Course

- Audience
- Prerequisites
- Agenda
- Purpose of the Course

Audience

- All Clinical Data Management personnel using Rave ***except*** for Clinical Programmers
- Other Development personnel using Rave

Prerequisites

- Intro to EDC
- EDC Business Process Training (online)
- EDC Study Specification course (online)

Agenda

Section 1: Introduction

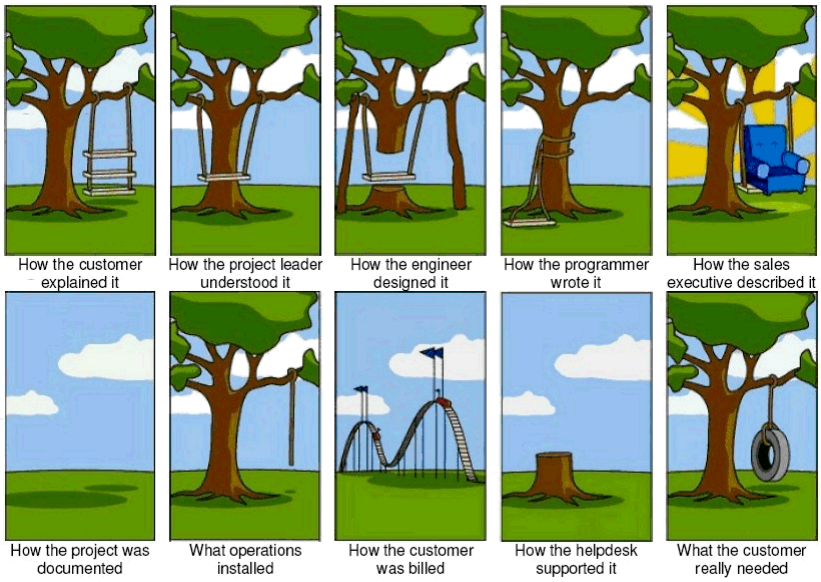
General introduction to the terminology, processes, and tools used to build EDC studies for Roche-sponsored clinical trials

Section 2: Building Studies

In-depth, hands-on experience with using study specification to build a study in Architect

Purpose of the Course

- Provide information that can be used to produce better study specifications for Roche-sponsored clinical trials that use Rave EDC, including:
  - terminology, processes, and tools used by Clinical Programmers
  - Roche standards, naming conventions, Clinical Programmer best practices, design considerations, and lessons learned
- Clinical programmers (CPs) rely on study specifications that are well written, precise, and thorough. By having hands-on experience with Architect in this course, you will better understand the necessity of properly-prepared study specifications.



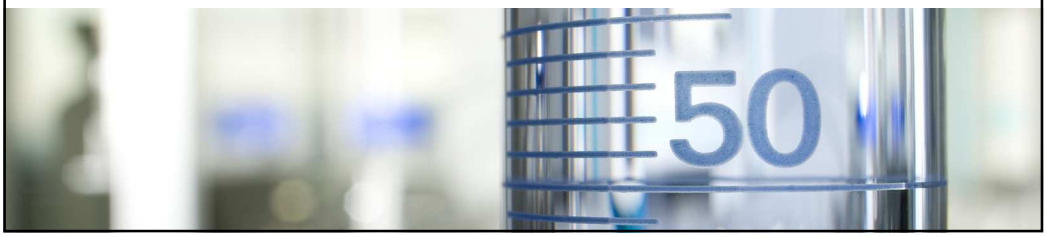
Housekeeping

- Where are the toilets?
- When will we take breaks?
- Will my mobile phone or PDA be ringing?
- When can I answer my emails?

Ground Rules

- ❑ Let's have fun
- ❑ Questions are welcome at all times
- ❑ Respectful and safe environment
- ❑ SDMs must attend the entire class to get credit

Section 1: Introduction



Objectives for Section 1

In Section 1, you will be able to:

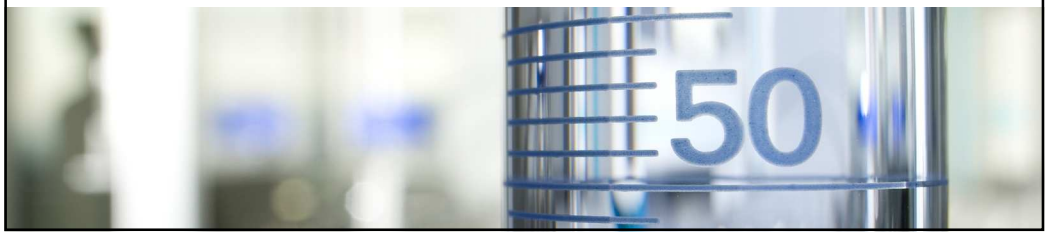
- ❑ Understand key study build concepts, such as:
  - Rave Architect functionality
  - Roche study build process
  - Architect project components
  - Roche Global Library Volume
- ❑ Launch Architect and explore a sample project:
  - Projects, drafts, versions, and studies
  - Draft items, including forms and fields, folders, dictionaries, matrices, edit checks, custom functions, derivations, and restrictions

Lessons To Complete in Section 1

- ❑ Lesson 1: Introduction to this Course
- ❑ Lesson 2: Key Rave Architect Concepts
- ❑ Lesson 3: Getting Started with Architect
- ❑ Lesson 4: Exploring Items in a Draft
- ❑ Lesson 5: Exploring Fields on a Form

Lesson 1: Introduction to this Course

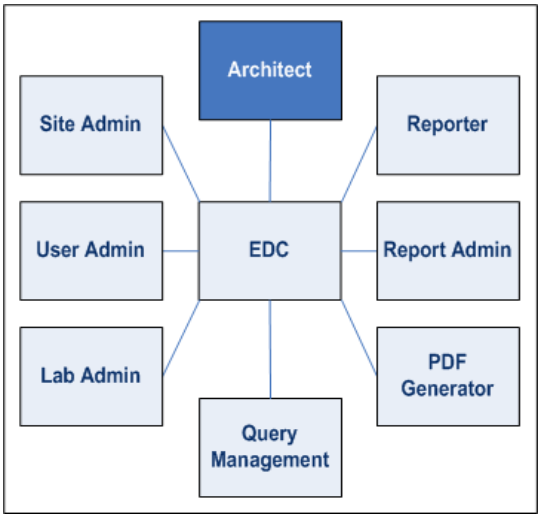
Participant Guide page 2



Topics To Cover in Lesson 1

- ❑ About the Rave Architect Module
- ❑ What You Will Learn in the Architect Lite Training
- ❑ Sample Project Used in this Training

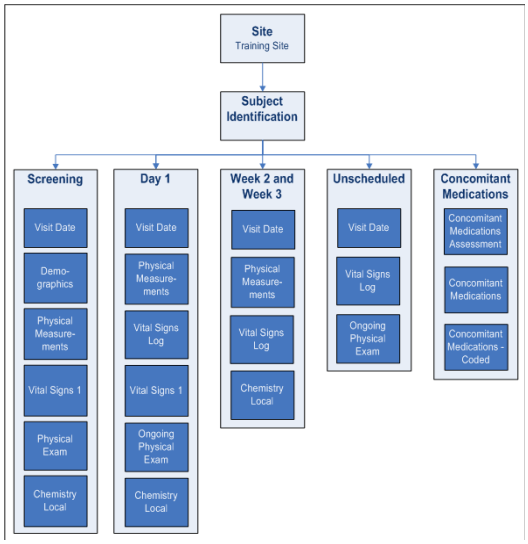
Medidata Rave Architect Module



Medidata Rave Architect Module

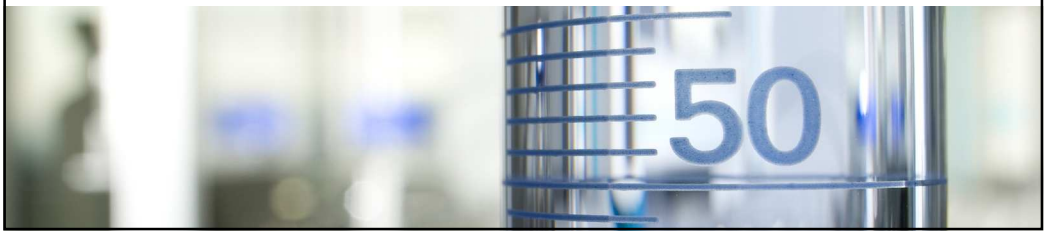
- Software that Roche Clinical Programmers use to build studies in Rave
- Provides Clinical Programmers with a comprehensive, full-featured development environment in which to:
  - build electronic case report forms (eCRFs), folders, and matrixes
  - design all the elements on an eCRF, including fields, variables, data entry controls, labels, lists, help text, SAS labels, and so on
  - write data logic for eCRF data, including data entry checks and edit checks that can trigger queries, stickies, and other actions
  - specify role-based view and entry restrictions to EDC data
  - copy standard elements from Roche's Global Volume
  - create and manage separate development, testing, production, and training environments

Folders & eCRFs in the Demo Study



Lesson 2: Key Rave Architect Concepts

Participant Guide page 6



Topics To Cover in Lesson 2

- ❑ Study Build Process
- ❑ Clinical Programmer Responsibilities
- ❑ Study Specifications for Building Studies
- ❑ Relationships Between Study Specifications and Finished Studies in the Rave

Clinical Programmer Responsibilities

- ❑ 'CASA' is a system role within Rave.
- ❑ Formerly, Clinical Programmers (CPs) at Genentech had the job title of Clinical Applications Systems Analyst (CASA)
- ❑ Responsibilities
  - Technical support for Clinical Data Management in:
    - clinical database development
    - edit check programming
  - Serve as a study configuration expert to Clinical Data Management and implement the study configuration in Rave
- ❑ If you understand what a CP sees in Architect:
  - You will understand more about why and how the study specifications are written
  - You will be able to more effectively develop and review study specifications for your study

Study Specifications

- ❑ Study Configuration Document (SCD)
- ❑ Mock eCRFs / eCRF Help Text / Source Document Verify (SDV)
- ❑ View & Entry Restrictions (VER)
- ❑ Study Logic and Check Specifications (SLACS) / Visit Form Matrix (VFM)

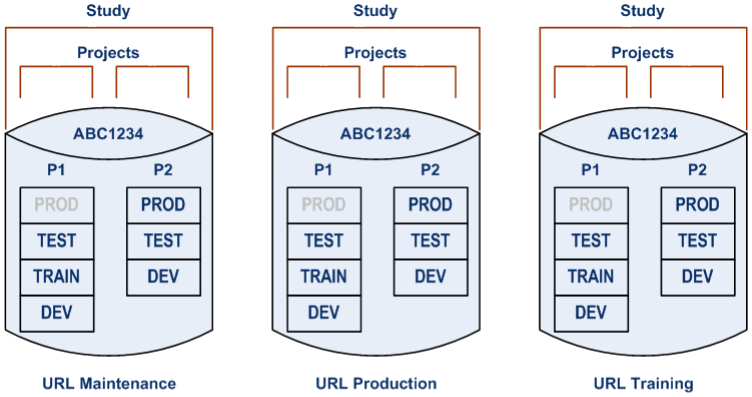
  Note: Unused functionality within Rave Architect is still captured as blank items in the study specifications documents (example: Open, Close, Start, and Stop functions of Subject Calendar).

You have electronic copies of the SCD, VER, SLACS, and VFM on your system. The Mock eCRFs and SDV are printed handouts

Study Specifications

Item in Architect	Specified in
Forms	Mock eCRFs
Edit Checks	SLACS
Folders	Visit/Form Matrix
Custom Functions	SLACS and Custom Functions Specification Document
Patient calendar	Visit calendar in Study Configuration Document
Primary matrix	Visit/Form Matrix
V&E Restrictions	View and Entry Restrictions
Automated Sticky Notes	SLACS
Comments on Audit Trail screen (for Labs only)	SLACS
TMS integration	Study Configuration Document and Integration Custom Functions Specification
Analyte	Mock eCRFs

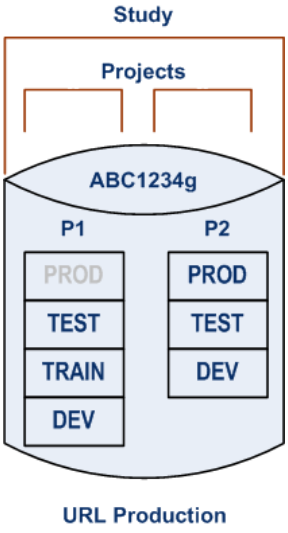
Rave URLs Used at Roche



Other URLs:

- DEV – Product evaluation, discovery and pre-validation activities, PD Informatics
- TEST – PD Informatics
- Sandbox – Play Area

Rave Projects Used at Roche



Lesson 3: Getting Started with Architect

Participant Guide page 8

Topics To Cover in Lesson 3

- ❑ Launching Architect
- ❑ Architect Main Screen
- ❑ Navigating Architect
- ❑ Projects and Associated Components
- ❑ Roche's Global Library Volume
- ❑ Exploring the Sample Project
- ❑ Hands-On Exercise

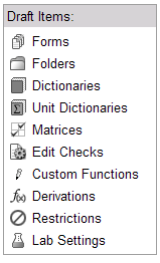
Rave Startup in the Training Environment

- Training URL: https://rochetrn563.mdsol.com
- Login Credentials (***case sensitive***):

Credential	Pattern	Example
User	casaal_*n*	casaal_1
Password	password3	password3

Navigation Tips

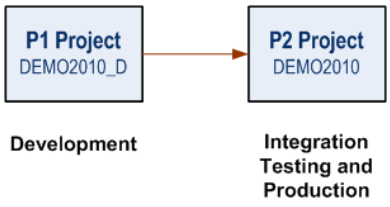
- Always use Rave buttons, tabs, and links to navigate the Rave interface.
  - Do not use the buttons (Back, Forward, Stop, Refresh, or Home) or menu commands in your Web browser.
  - If you use any of the browser buttons, Rave will prompt you to re-enter your password before you can resume using Rave.
- Time-outs occur after a period of inactivity (45 minutes or more).
  - Any unsaved data is lost when a time-out occurs
  - You may be prompted to log in again
- The sidebar provides easy navigation to tasks or components that are relevant in the current context. Example: on the Draft page, the sidebar shows the types of items in the current draft.



Projects

- Contain the drafts, environments, subject search configurations, and copy source definitions
- Drafts contain
  - eCRFs
  - folders
  - matrices
  - edit checks
  - other draft-specific elements
- When Clinical Programmers begin constructing a new EDC study for a Roche-sponsored clinical trial, they start by creating a new project in Architect

P1 and P2 Projects



- P1
  - Initial project
  - Used for all system development
- P2
  - Used for integration installation testing (also called P2 testing)
  - Used for production

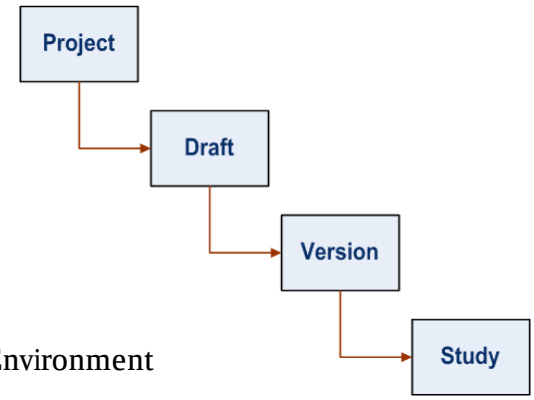
Environments

- A partitioned instance in the database for a particular purpose
- Name describes the context in which it is used
- Environments at Roche:

Name	Description
DEV	Where Clinical Programmers create and configure EDC studies
TEST	Where PDMs and project teams conduct online eCRF reviews and formal user acceptance testing (UATs)
TRAIN	Where system users are trained
PROD	Where the Rave EDC study is deployed live. Contains real clinical data gathered from actual study participants and is submitted to the FDA.

Project Components

1. Create a Project
2. Create a Draft
3. Publish a Version
4. Push to a Study Environment



Drafts

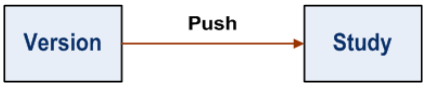
- Form of the project that Clinical Programmers can view and edit
- Contains all the elements of a study that a Clinical Programmer can develop—eCRFs, fields and variables, folders, data validations, and so on
- Clinical Programmers devote a lot of time to working on drafts of a project
- A project can contain multiple drafts

Versions

- Saved snapshot of a draft, including all of the study elements that were configured in the draft at the time the version was created
- Shows how the study elements created in Architect will be displayed to users in the EDC module
- CRFs are associated with a specific version. The CRF version number is displayed at the bottom left of the eCRF screen.
- A project can contain multiple versions
- To create a new version, you publish a draft

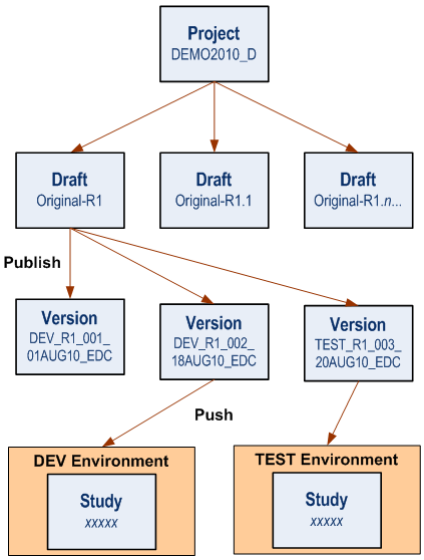
Studies

- Electronic casebook that users can access and navigate in the Rave EDC module
- To create a new study that can be used in the Rave EDC module, you push a version to an environment



- Once a study is available in that environment, authorized users can access the study and enter data

How These Components are Related



Roche's Global Library Volume

- Extensive collection of standard EDC study components
- Called ***Global Volume Integrated (GVI)*** in Architect
- Rather than build entire studies from scratch, Clinical Programmers can quickly assemble studies using pre-built components copied from the Global Library Volume
- Using elements in the global library adds efficiencies to study builds
- Using unchanged global standards eliminates the requirement to UAT these elements

Hands-on Exercise for Lesson 3

- See page 21
- Answer the question in step 3

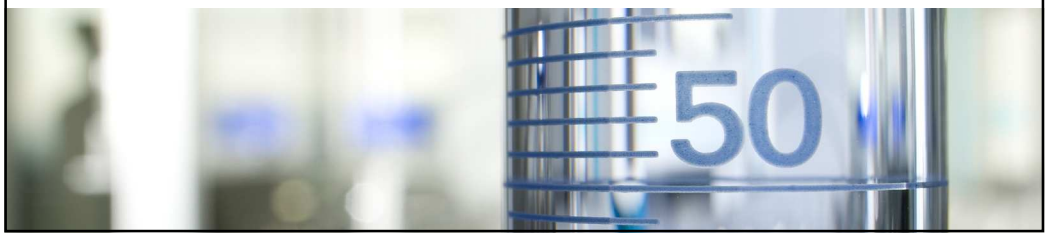
Answer for Hands-On Exercise

The Primary Form for this draft is:

- Subject Identification form

Lesson 4: Exploring Items in a Draft

Participant Guide page 22



Topics To Cover in Lesson 4

- ❑ Selecting Items to Explore
- ❑ Viewing Forms
- ❑ Viewing Folders
- ❑ Viewing Dictionaries
- ❑ Viewing Matrices
- ❑ Viewing Edit Checks
- ❑ Viewing Custom Functions
- ❑ Viewing Derivations
- ❑ Viewing Restrictions
- ❑ Hands-On Exercise

Hands-On Exercise for Lesson 4

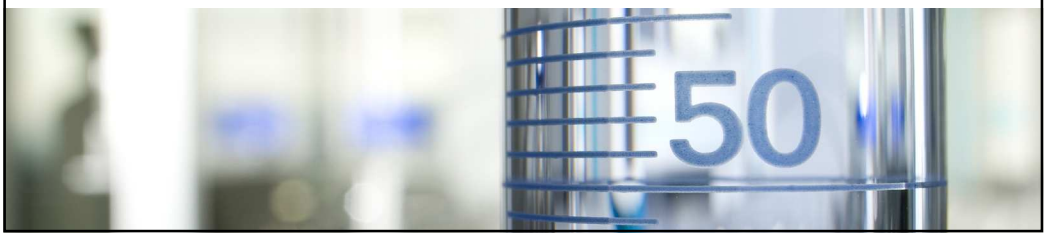
- ❖ See page 38
- ❖ Answer the questions for each step

Answers for Lesson 4 Hands-On Exercise

1. The form with the form OID value of PE2 is the Ongoing Physical Exam form.
2. The folder names for which no target values have been assigned are: Unscheduled and Concomitant Medications.
3. The folders that have the Visit Date form specified in the Primary Matrix are: Screening, Day 1, Week 2, and Week 3.

Lesson 5: Exploring Fields on a Form

Participant Guide page 39



Topics To Cover in Lesson 5

- ❑ About Fields
- ❑ Selecting a Field on a Form
- ❑ Previewing Fields on a Form
- ❑ Viewing:
  - Field Properties
  - Variable Settings for a Field
  - Field Help Text
  - Field Edit Checks
  - Verification and Review Settings
  - View and Entry Restrictions
  - Edit Checks
  - Derivations
- ❑ Hands-On Exercise

Hands-On Exercise for Lesson 5

- ❖ See page 52
- ❖ Answer the questions for steps 1 and 2

Answers for Lesson 5 Hands-On Exercise

1. Answers for COLD field:
   a. The correct variable format is: dd- MMM- yyyy
   b. No, it is not configured as a log field
   c. Yes, help text exists for this variable
   d. Two (2) field edit checks are configured for this field
2. Answers for TEMP field:
   a. True. The edit checks are system autoqueries
   b. Yes, this variable will be SDV'd
   c. False. There are no view restrictions on this field.
   d. False. There are no entry restrictions on this field.
3. Answers for TEMPU field:
   a. False. There are no edit checks for this field.
   b. True. There are view restrictions on this field.
   c. False. There are no entry restrictions on this field.

What Was Covered in Section 1

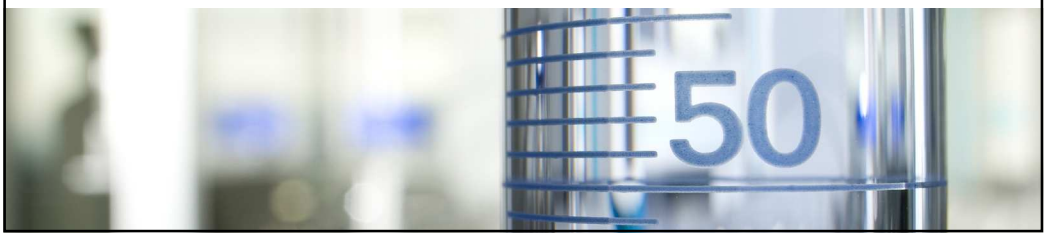
- ❑ Overview of Rave Architect functionality
- ❑ Architect project components (projects, drafts, versions, and studies)
- ❑ Roche Global Library Volume
- ❑ Launching and Navigating Architect
- ❑ Exploring a sample project
- ❑ Exploring draft items, including forms and fields, folders, dictionaries, matrices, edit checks, custom functions, derivations, and restrictions

Where To Go From Here

- CDM Staff Continuing Onto Section 2
  - Return after the break
- Other Participants
  - Training is complete
  - Please sign the group training record
  - Please complete the Course Survey:
    http://www.surveymonkey.com/s/CDM-PTM_Post-Training_Reaction_Survey
    - ***Course Name**: Architect Lite Session 1*
    - ***Please know that should you have issues you should delete your cookies.***

Section 2: Building a Study

Participant Guide page 55



Lessons To Complete in Section 2

- ❑ Lesson 6: Working with Projects, Environments, and Drafts
- ❑ Lesson 7: Copying Items from the Global Library
- ❑ Lesson 8: Building Forms
- ❑ Lesson 9: Creating Folders
- ❑ Lesson 10: Creating Matrices
- ❑ Lesson 11: Testing Your Progress
- ❑ Lesson 12: Defining Restrictions
- ❑ Lesson 13: Defining Edit Checks
- ❑ Lesson 14: Using Custom Functions
- ❑ Lesson 15: Finishing Touches

Objectives for Section 2

In Section 2, you will learn how to:

- ❑ Create and configure projects, environments, and drafts
- ❑ Copy items from the Global Library
- ❑ Build forms for a study
- ❑ Configure folders and matrices for a study
- ❑ Publish a version, push it to a study, and see the results in the EDC Module
- ❑ Configure restrictions for forms and fields
- ❑ Configure edit checks and use custom functions
- ❑ Creating a draft in P2 from your P1 project

Lesson 6: Projects, Environments, and Drafts

Participant Guide page 56

Topics To Cover in Lesson 6

- ❑ Working with Projects
- ❑ Working with Environments
- ❑ Working with Drafts
- ❑ Hands-On Exercise

Ways to Create a Draft

- Create a Blank Draft
  - Populate Using the Copy Wizard
- Copy from Pre-Existing Project Version
- Architect Loader
- Copy from Global Library (not used at Roche)

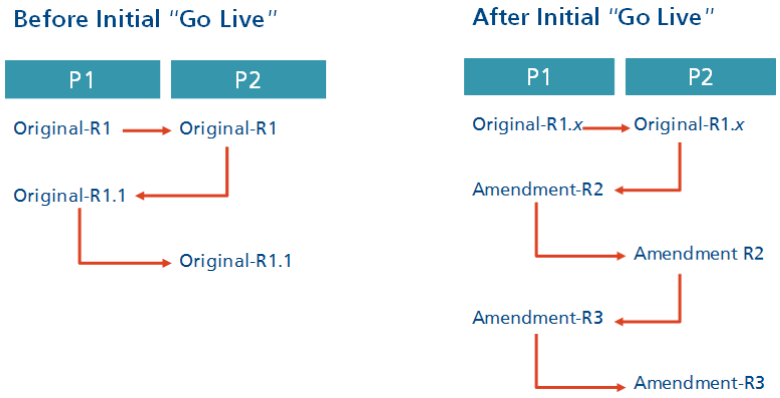
Naming Conventions for Drafts

Status-Release

where

- ***Status*** = status of the protocol from which it was built
  - Original
  - Amendment
- ***Release*** = release number
  - First release: R#, where # is a sequential number
  - Subsequent releases if there is a change in a study after it goes to P2 but before go live: R#.#, where #.# is a sequential number
  - The decimal point increments the number of times this has happened, such as R1.1, R1.2, and so on

How Drafts Are Named

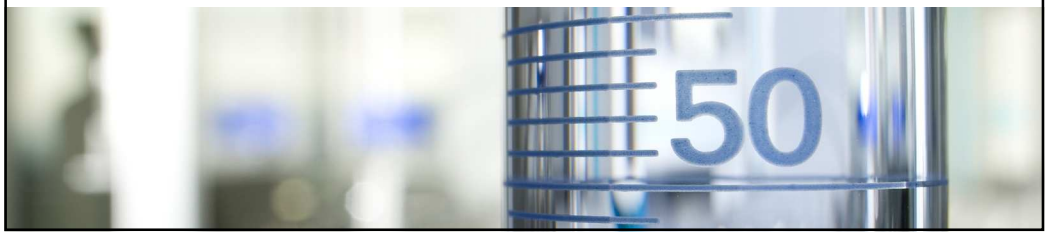


Demo and Hands-On Exercise for Lesson 6

❖ See page 65

Lesson 7: Copying Items from the Global Library

Participant Guide page 66



Topics To Cover in Lesson 7

- ❑ About Copying from the Global Library
- ❑ Adding the Global Library as a Copy Source
- ❑ Running the Copy Wizard
- ❑ Searching for an Item in the Global Library
- ❑ Arranging eCRFs on the Forms Screen
- ❑ Hands-On Exercise

Leader Demo

Running the Copy Wizard

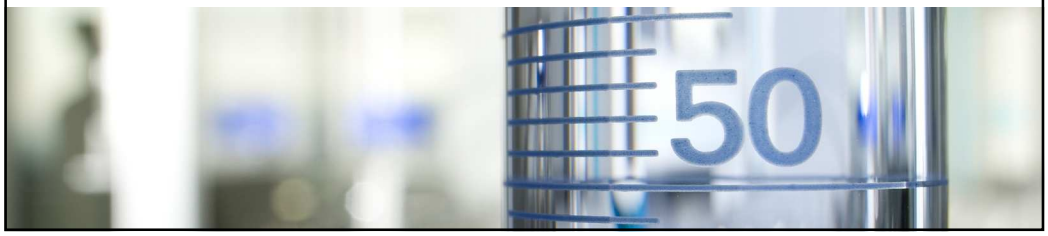
- ❑ Copy a form
- ❑ How to check for the fields
- ❑ Copy an edit check
- ❑ Copy a derivation

Hands-On Exercise for Lesson 7

- ❖ See page 71

Lesson 8: Building Forms

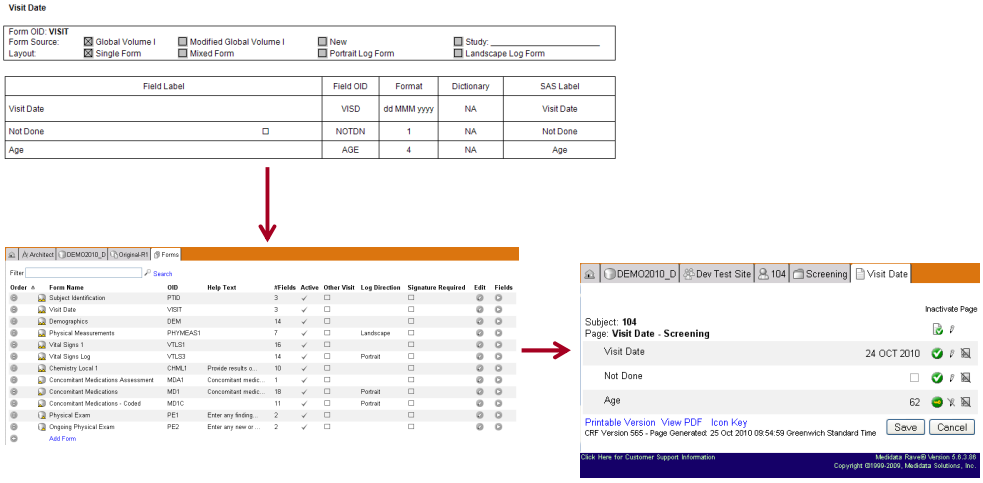
Participant Guide page 73



Topics To Cover in Lesson 8

- ❑ About Forms (eCRFs)
- ❑ Copying Forms from the Global Library
- ❑ Creating Forms
- ❑ Editing Form Settings
- ❑ Deleting Forms
- ❑ Adding Fields to a Form
- ❑ Working with Dictionaries
- ❑ Working with Variables
- ❑ Specifying Help Text for Forms and Fields
- ❑ Hands-On Exercise

Mock eCRFs and Forms



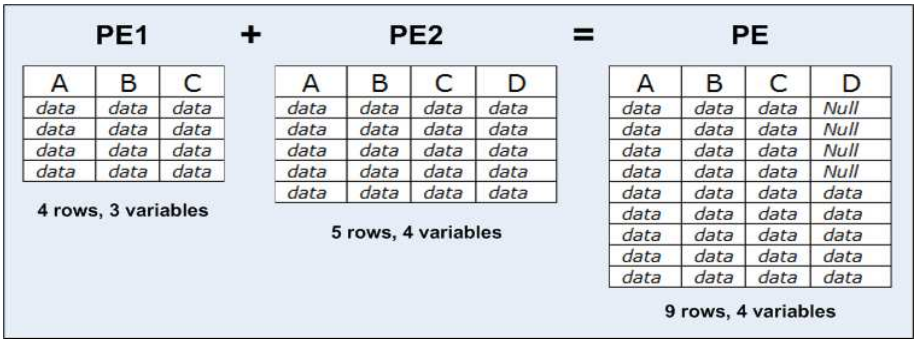
Naming Conventions for Forms

- Unique, descriptive name that indicates its purpose
- Form names are specified in the Mock eCRFs
- Form OIDs must be unique and are no more than six characters long (including digits)
- Studies must use standard form OIDs—the form OIDs cannot be changed
- Forms with a similar purpose have the same root OID
- For coded forms, the Form OIDs have the same root as the form it is coding. Example: AEDEC

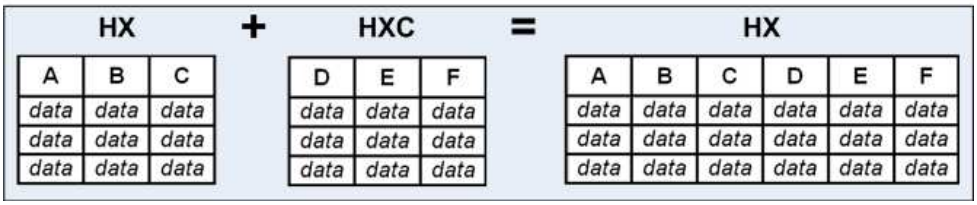
Dataset Conversion and Form OIDs

- When the SAS datasets are extracted from Rave, the datasets are combined under the following conditions:
  - a Form OID is used more than one time, and
  - the value ends in either a number or 'C' (for coded forms)

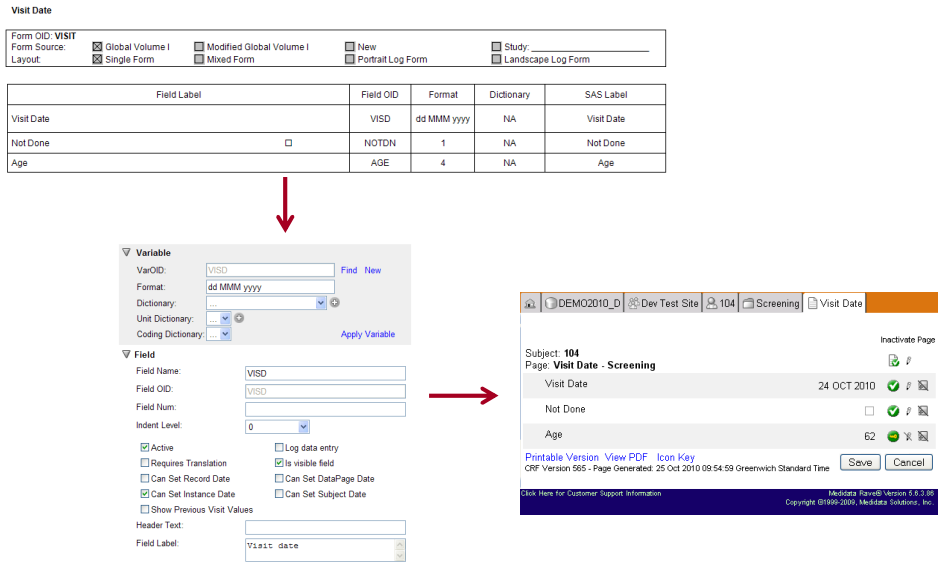
Form OIDs Ending in a Number



Form OIDs and Coded Forms (Ending in a “C”)



Mock eCRFs and Fields

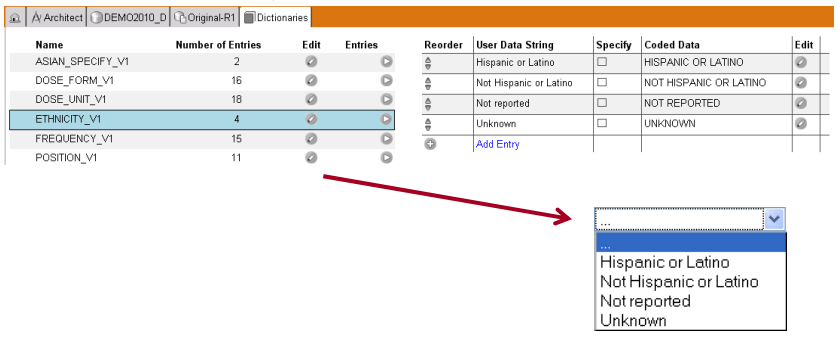


Naming Conventions for Fields

- Unique, descriptive name that indicates its purpose
- Field names are specified in the Mock eCRFs
- For field OIDs, Roche follows standard naming conventions
- Field OID is the SAS name

Mock eCRFs and Dictionaries

Ethnicity ↓ Hispanic or Latino ↓ Not Hispanic or Latino ↓ Not reported ↓ Unknown	ETHNIC	$22	ETHNICITY_ V1	Ethnicity



Naming Conventions for Dictionaries and Entries

DescriptiveName_V#

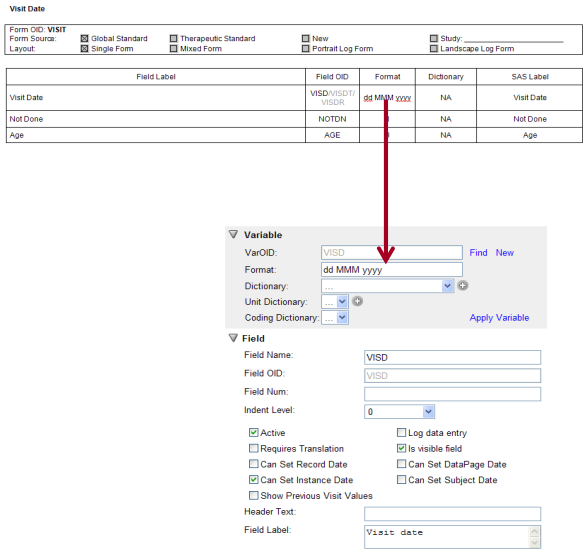
where

- ***DescriptiveName*** is a brief name that indicates the purpose of the dictionary
- ***_V#*** is the version number of the dictionary being used
  - For the Global Volume, this is always zero (0)
  - If greater than zero, the dictionary represents a subset of what's in the Global Volume

Dictionary entries use:

- **User Data Strings**: Mixed case and any keyboard characters. The values displayed on the screen in the EDC module in Rave.
- **Coded Values**: All-uppercase characters and spaces. Avoid special characters of any kind in coded values, including punctuation. Coded Values are what is stored. Coded values are what will be seen on Data Listing Reports, J-Review queries, or any other report that accesses the data.
- In most cases, the text value of the user data string and the coded value is the same. The exception is QOL questionnaires.

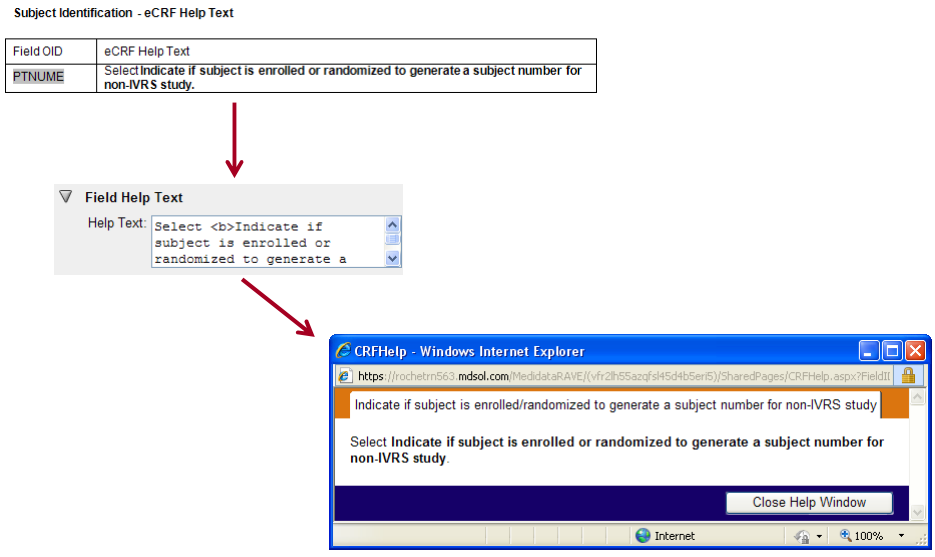
Mock eCRFs and Variables



Naming Conventions for Variables

- Variable OIDs and Field OIDs are usually the same
- Exception: SAE Report Summary form
- Variable OIDs must be unique when two (or more) fields on different forms have the same field OID but
  - different formats, or
  - different dictionaries
- Variable settings are specified in the Mock eCRFs

Mock eCRFs and Help Text



Leader Demo

- ❑ Build PE1 Form
- ❑ Build PE2 Form as described in the exercise
- ❑ Edit the MD1 Form

Hands-On Exercise for Lesson 8

- ❖ See page 88

Lesson 9: Creating Folders

Participant Guide page 91

Topics To Cover in Lesson 9

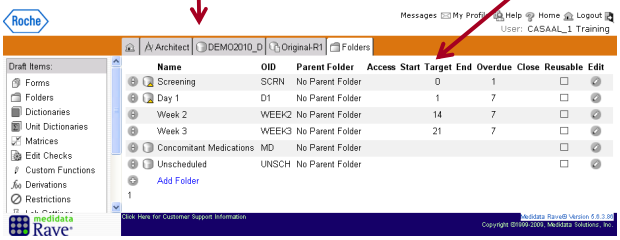
- ❑ About Folders
- ❑ Copying Folders from the Global Library
- ❑ Creating Study-Specific Folders
- ❑ Hands-On Exercise

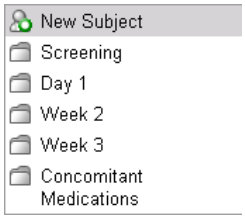
Visit Form Matrix, Visit Calendar, and Folders

PRIMARY VISIT FORM MATRIX

eCRF Rave Display Order	FOLDER OID	FOLDER NAME	Subject Level	SCRN	D1	W2	W3	MD
				Screening	Day 1	Week 2	Week 3	Concomitant Medications
	Form Name	Form OID						
1	Visit Date	VISIT		X	X	X	X	

Visit Folder	Start	Target	End	Overdue
Screening		0		1
Day 1		1		7
Week 2		14		7
Week 3		21		7





Naming Conventions for Folders

- Unique, descriptive name that indicates its purpose
- Specified in the Visit / Form Matrix
- Roche uses the following guidelines:
  - Maximum length of 50 characters
  - Can have both upper and lower case characters.
  - Cannot contain underscores, but may contain spaces or dashes. No more than one space or one dash is allowed between text characters. No spaces should be placed around dashes. No other special characters are allowed.
  - If word abbreviations are required to conform to the 50 character limit, use abbreviations that are recognizable to others.
    - Refer to the Table of Standard Abbreviations and a Table of Standard Folder Names and Folder OIDs in the Biometric Global Process Library.
    - Double-check the standard to see whether the folder name has already been approved.

Naming Conventions for Folder OIDs

- Folder object IDs (OIDs) should be unique, brief, and descriptive
- Specified in the Visit / Form Matrix
- Roche uses the following guidelines for folder OIDs:
  - Maximum of 10 characters long
  - All uppercase text
  - Cannot contain underscores, spaces, dashes, or any other special characters
  - General conventions for new folder OIDs:
    - Use the first letter of the event name followed by the number of the event.
      Example: Week 1 Day 3 would be W1D3
    - If the event is a single word it can be abbreviated.
      Example: Unscheduled would be UNSCH
    - If the event is multiple words, use the first letter of each word.
      Example: End of Study would be EOS
- Refer to the Table of Standard Folder Names and Folder OIDs in the Biometric Global Process Library.

Follow Along With the Instructor

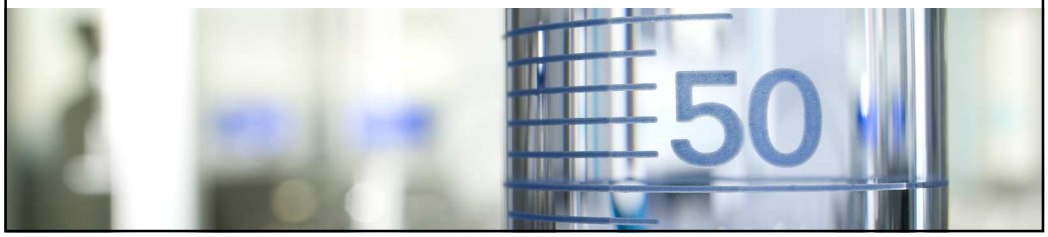
- Copy folders from GVI
- Copy folders from Demo2010_D
- Edit calendar settings per specifications

Hands-On Exercise for Lesson 9

- See page 95

Lesson 10: Creating Matrices

Participant Guide page 96



Topics To Cover in Lesson 10

- ❑ About Matrices
- ❑ Copying Matrices from the Global Library Volume
- ❑ Modifying a Matrix
- ❑ Creating a Matrix (Unscheduled Visit)
- ❑ Hands-On Exercise

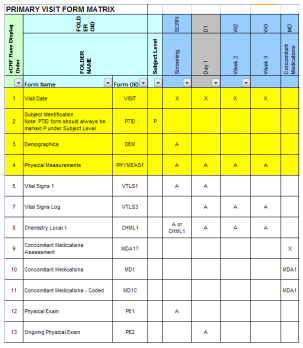
Types of Matrices

Matrix	Description
Primary Matrix	Default matrix that is automatically created when a new subject is added
Add Event Matrix	Allows for adding more folders upon request, such as for unscheduled visits. Sites can add visits manually from a drop-down list
Master Matrix	Used to generate blank eCRFs and PDF files for annotated CRFs. Specifies the display order and identifies subject-level forms. The GVI includes a Master Form matrix.
Study-specific Matrix	Used when you have two different cohorts in a single study. Each cohort uses a different matrix. Strategy to minimize extraneous folders for each subject.

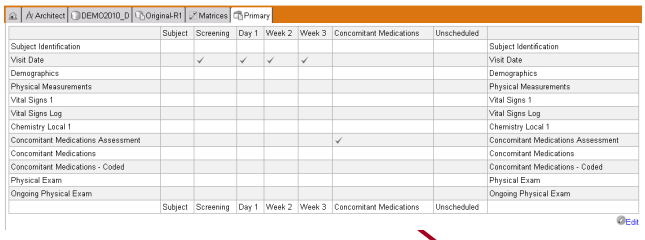
Standard Matrices at Roche

- ❑ DNA or RCR Research Sample Consent Change
- ❑ Early Term
- ❑ Master Form
- ❑ Optional Research Sample Consent Change
- ❑ Pregnancy Report
- ❑ Primary
- ❑ Unscheduled
- ❑ Unscheduled Local Labs

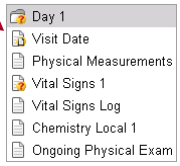
Visit Form Matrix and Matrices


PRIMARY VISIT FORM MATRIX

	Form Name	Form OID	Subject Level	Screening	Day 1	Week 2	Week 3	Concomitant Medications
1	Visit Date	VISIT		X	X	X	X	
2	Subject Identification Note: PTID form should always be marked P under Subject Level.	PTID	P					
3	Demographics	DEM		A				
4	Physical Measurements	PHYMEAS1		A	A	A	A	
5	Vital Signs 1	VTLS1		A	A			
7	Vital Signs Log	VTLS3			A	A	A	
8	Chemistry Local 1	CHML1		A or CHML1	A	A	A	
9	Concomitant Medications Assessment	MDA1?						X
10	Concomitant Medications	MD1						MDA1
11	Concomitant Medications - Coded	MD1C						MDA1
12	Physical Exam	PE1		A				
13	Ongoing Physical Exam	PE2			A			


Architect | DEMO2010_D | Original-R1 | Matrices | Primary

	Subject	Screening	Day 1	Week 2	Week 3	Concomitant Medications	Unscheduled	
Subject Identification								Subject Identification
Visit Date		✓	✓	✓	✓			Visit Date
Demographics								Demographics
Physical Measurements								Physical Measurements
Vital Signs 1								Vital Signs 1
Vital Signs Log								Vital Signs Log
Chemistry Local 1								Chemistry Local 1
Concomitant Medications Assessment						✓		Concomitant Medications Assessment
Concomitant Medications								Concomitant Medications
Concomitant Medications - Coded								Concomitant Medications - Coded
Physical Exam								Physical Exam
Ongoing Physical Exam								Ongoing Physical Exam
	Subject	Screening	Day 1	Week 2	Week 3	Concomitant Medications	Unscheduled	



Leader Demo

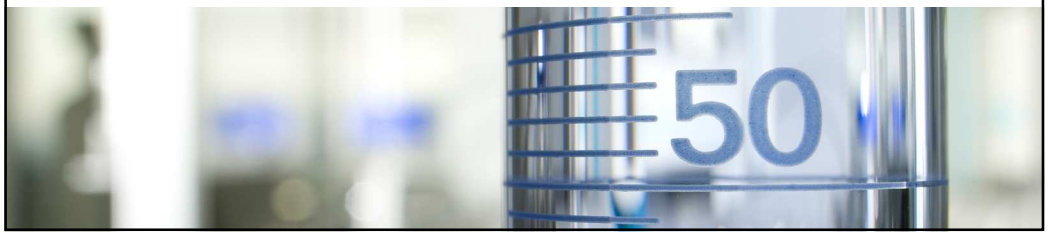
- Copy from Global Library Volume
- Editing the Draft Settings

Hands-On Exercise for Lesson 10

- See page 101

Lesson 11: Testing Your Progress

Participant Guide page 103



Topics To Cover in Lesson 11

- ❑ Publishing a Version from a Draft
- ❑ Pushing a Version to a Study
- ❑ Reviewing Your Results
- ❑ Making Subsequent Changes
- ❑ Hands-On Exercise

Naming Conventions for Versions

CCC_R***#.#***_***###***_***ddMMMyy_AAA***

Token	Description
CCC	Environment where the version will be pushed (DEV, TRN, TST, or PRD)
R*#.#*	Draft from which the version was published
###	Sequential number of versions published on that date
ddMMMyy	Date on which the version was created
AAA	Initials of the Clinical Programmer who created the version

Roles for Reviewing Your Results

- Go to the EDC module and review the results of a pushed study
- The CASA-1 role provides unrestricted access so that Clinical Programmers can see the behavior of all the fields they build for later testing
- Clinical Programmers can run a report to view role-based functionality.

Follow Along With the Instructor

- ❑ Create a Version

Hands-On Exercise for Lesson 11

- ❖ See page 110

Lesson 12: Defining Restrictions

Participant Guide page 111

Topics To Cover in Lesson 12

- ❑ About Restrictions
- ❑ Configuring View and Entry Restrictions for a Form
- ❑ Configuring Field Restrictions
- ❑ Testing Restrictions
- ❑ Hands-On Exercise

Roles and Restrictions

- **Role** - Set of permissions to perform certain actions and access particular data
- **Restriction** - Controls the degree to which a given EDC role has access to forms or field

Implications of Restrictions on Roles

- If a role is restricted from viewing a field or form, then users associated with that role:
  - will not see that form or field in the EDC module
  - will see blanks—instead of restricted fields—on reports in the Reporter module
  - will not see any columns representing restricted fields in the data set, or they will see blanks instead of the restricted fields
  - if a form contains restricted fields, and a form-level operation (such as a lock or freeze) is applied to the form, the operation does not apply to any restricted fields
- Default restrictions are included when you copy forms and fields from the Global Library Volume into your draft

View Entry Restrictions

View and Entry Restrictions Specifications

Roles used in study:
<List here all roles used in the study as defined in the study-specific Study Configuration Document (SCD) and incorporate throughout this document, changing those roles below as necessary. For Role definitions, please see the study SCD.>

Note: Do not list Instruction Text (BLUE1, BLUE2, etc.) for non-log forms; only list Instruction Text for log forms.

STANDARD FORM NAME and FIELD OID	Form OID	VIEW Restriction	ENTRY Restriction
Chemistry Local 1	**CHML1**		Batch Upload, INT-1, INT-DS1
TMPTC	CHML1		CRC-1I, PI-1I
PGREP1	CHML1	Batch Upload, CRA-1, CRA-1A, CRC-1I, CTM-1, INT-1, INT-DS1, PI-1I, VIEW-1S	
Concomitant Medications Assessment	**MDA1**		Batch Upload, INT-1, INT-DS1
Concomitant Medications	**MD1**		Batch Upload, INT-1, INT-DS1
MDTYP	MD1	CRC-1I, PI-1I, VIEW-1S	
Concomitant Medications – Coded	**MD1C**	CRA-1, CRA-1A, CRC-1I, CTM-1, PI-1I, VIEW-1S	Batch Upload, INT-DS1
Demographics	**DEM**		Batch Upload, INT-1, INT-DS1
AGEIC, AGEU	DEM	CRC-1I, PI-1I, VIEW-1S	

Settings on the Restrictions Screen

Settings on the Restrictions screen include:

- ❑ whether the field requires verification (SDV)
- ❑ whether the field requires manual review (rare)
- ❑ which EDC role(s) can view or edit the field
- ❑ default value for field

Leader Demo

- ❑ Form Restrictions
- ❑ Publish and Push

Hands-On Exercise for Lesson 12

- ❖ See page 118

Lesson 13: Defining Field Edit Checks

Participant Guide page 119

Topics To Cover in Lesson 13

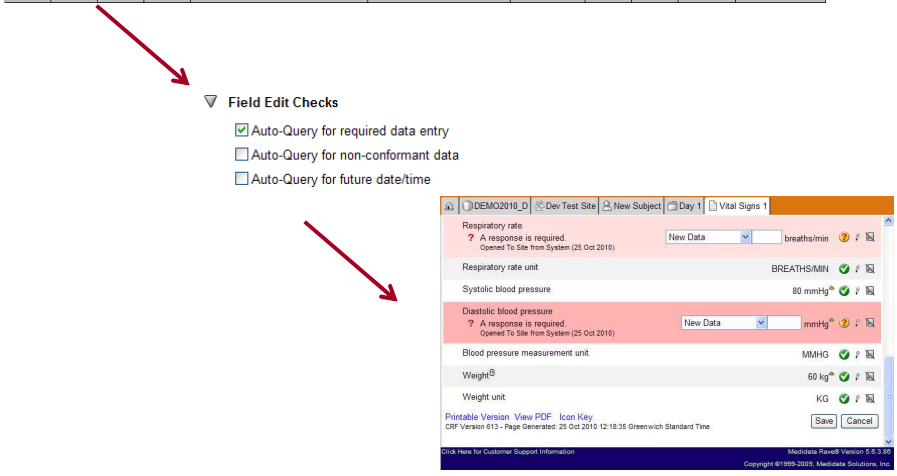
- ❑ About Edit Checks
- ❑ Configuring Field Edit Checks
- ❑ Hands-On Exercise

Field Edit Checks

- Built-in mechanisms that Rave provides for catching common data entry errors in fields
- Trigger system-generated queries for an individual field based on an error condition
- Examples:
  - no data was entered in a required field
  - entered data was non-conformant
  - future date was entered in a date field
  - data is out of a specified range
- Specified in the Study Logic and Check Specifications (SLACS)

SLACS and Field Edit Checks

Standard/ Study Specific	Form Name	Form OID	Line #	Check Description	Message	Check Action	Action Field	Resp Box and Manual Close	GVI Group Comment	CheckName
Global Standard	Vital Signs 1	VTLS1	30	**Were any vital signs collected at this visit? (VTLREC) is Yes**, then response to **Diastolic blood pressure (BPD)** must be provided.	A response is required.	Open Query	BPD			GE_VTLREC_YES_BPD_BLANK



Types of Field Edit Checks

- Auto-Query for required data entry
- Auto-Query for non-conformant data
- Auto-Query for future date/time
- Auto-Query for data out of range
- Mark non-conformant data out of range (not used at Roche)

Leader Demo

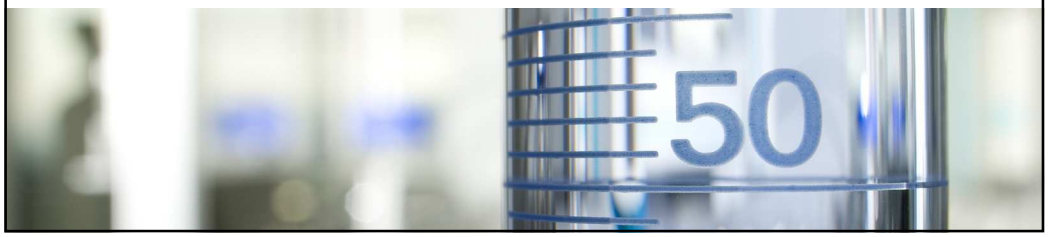
- Field Edit Check

Hands-On Exercise for Lesson 13

❖ See page 122

Lesson 14: Edit Checks and Custom Functions

Participant Guide page 123

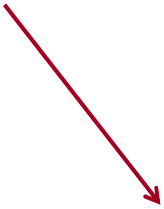


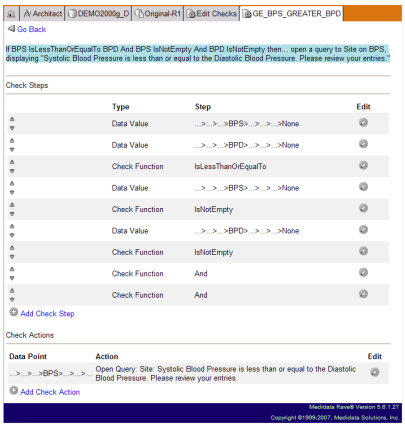
Topics To Cover in Lesson 14

- ❑ Working with Edit Checks
- ❑ Working with Custom Functions
- ❑ Hands-On Exercise

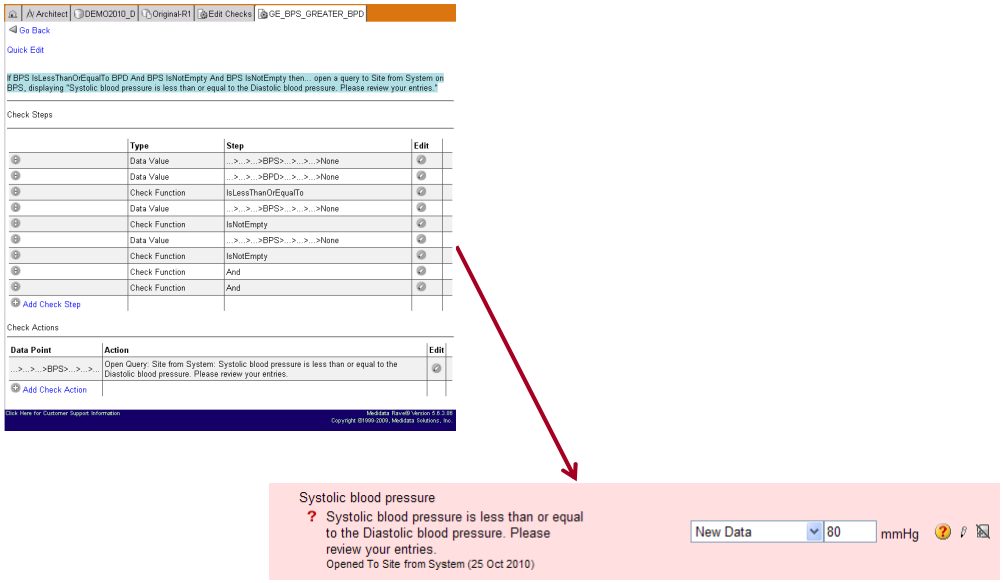
SLACS and Edit Checks

Standard/ Study Specific	Form Name	Form OID	Line #	Check Description	Message	Check Action	Action Field	Resp Box and Manual Close	GVI Group Comment	CheckName
Global Standard	Vital Signs 1	VTLS1	34	**Systolic Blood Pressure (BPS)** must be greater than **Diastolic Blood Pressure (BPD)**.	Systolic Blood Pressure is less than or equal to the Diastolic Blood Pressure. Please review your entries.	Open Query	BPS			GE_BPS_GREATER_BPD





Edit Checks and EDC



Naming Conventions for Edit Checks

- Edit check names are:
  - long and complex
  - encoded with meaning so that, when you're looking through a long list of edit checks in Architect, you can learn—at a glance—a lot about an edit check without needing to open it
- Syntax

  `[Scope][Type][Operation][C]_`*`SASName`*`_Description[_FolderSuffix]`

  where the *SASName* is the field OID
- See page 124 in your ***Participant Guide***

Check Steps and Check Actions

- Check Step
  - Provide the logic statement for the edit check
- Check Action
  - What the edit check will do if the conditions are met in the check step

Problems with Edit Check Descriptions

- Errors, Omissions, and Ambiguities
- Syntax, Logic Errors
- Example: “Date of Exam must be before Visit Date at Screening”
- What’s wrong?
  - No field OIDs
  - No folder OID
  - No form OID for a cross form check
  - The check description and message conflict slightly
  - It is not clear what should happen if the dates are the same
  - No check name provided

Types of Custom Functions

- standard and study-specific custom functions
  - Copied from the Global Library Volume or standard project, along with the edit checks that fire them
  - Defined in the study's custom function specification document
- integration custom functions
  - Developed by Informatics and customized as needed to fit study-specific requirements
  - Defined in the study's integration custom function specification document

About Custom Functions

- Custom functions are called through the edit check engine
- They can:
  - provide additional logic as part of the check steps, or
  - they can provide more refined actions as part of the check actions, or
  - both
- Custom functions are written in C Sharp code
- One of the edit checks you will copy in has a custom function attached to it

Leader Discussion

- ❑ In-Fix Notation
- ❑ Edit Check Logic
- ❑ Write PE Edit Check (Optional)

Hands-On Exercise for Lesson 14

- ❖ See page 136
- ❖ Please answer the questions

Answers for Lesson 14 Hands-On Exercise

Answers for Question 1:

a. True

b. False

c. False

d. False (Add Form Action)

Answers for Question 5:

a. PE2

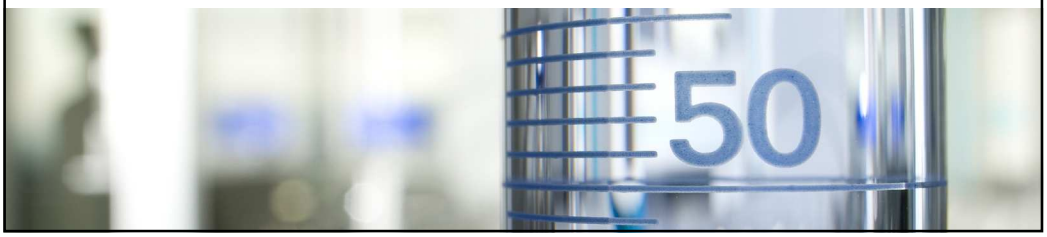
b. Open a query

c. Study-specific edit check

d. Action field (this should be PED)

Lesson 15: Finishing Touches

Participant Guide page 139



Topics To Cover in Lesson 15

- ❑ Creating a Draft in P2 from Your P1 Project
- ❑ Making Changes Late in the Development Cycle
- ❑ Hands-On Exercise

Leader Demo

- ❑ Creating a Draft in a P2 project

Hands-On Exercise for Lesson 15

- See page 141

What Was Covered in Section 2

- ❑ Configuring projects, environments, and drafts
- ❑ Copying items from the Global Library Volume
- ❑ Configuring draft items, such as forms, folders, matrices, dictionaries, restrictions, edit checks, and custom functions
- ❑ Configuring fields on a form, including variables, help text, restrictions, and other settings
- ❑ Publishing a draft to a version, pushing a version to a study, and reviewing the results of a study in the EDC module
- ❑ Creating a draft in P2 from your P1 project

Where To Go From Here

- ❑ Training is complete
- ❑ Please sign the group training record
- ❑ Please complete the Course Survey:
  http://www.surveymonkey.com/s/CDM-PTM_Post-Training_Reaction_Survey
  - *Course Name: Architect Lite Session 1 and Architect Lite Session 2*
    - ❖ *Should you have issues, you should delete your cookies.*

Roche

We Innovate Healthcare