



# Intelligent build from protocol to EDC

## CDISC US Interchange – COSA Workshop

**Sarah Jamal**

Principal Solutions Consultant

Oracle Life Sciences, Centre of Excellence



# Agenda

---

## **Introduction**

Working with OpenStudyBuilder

Clinical Automation

## **Demonstration**

Metadata Extraction POC for Schedule of Activities

EDC Study Build Automation

## **Moving forward**

Metadata extraction potential applications

Synthetic data generation

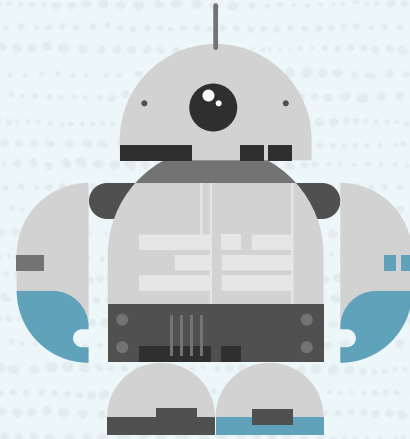
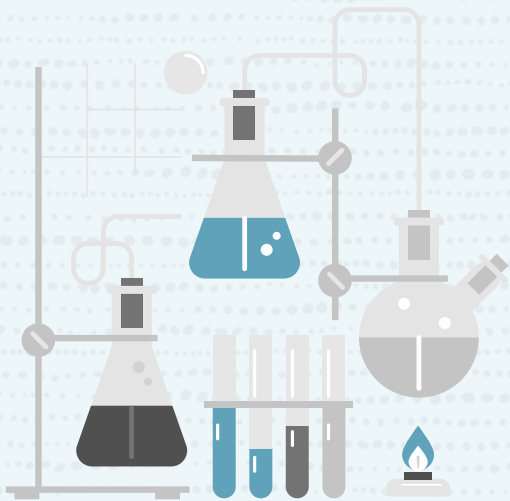
# Introduction

---

Working with OpenStudyBuilder  
Clinical Automation

**Oracle Life Sciences,  
Center of Excellence**

*Innovation projects*



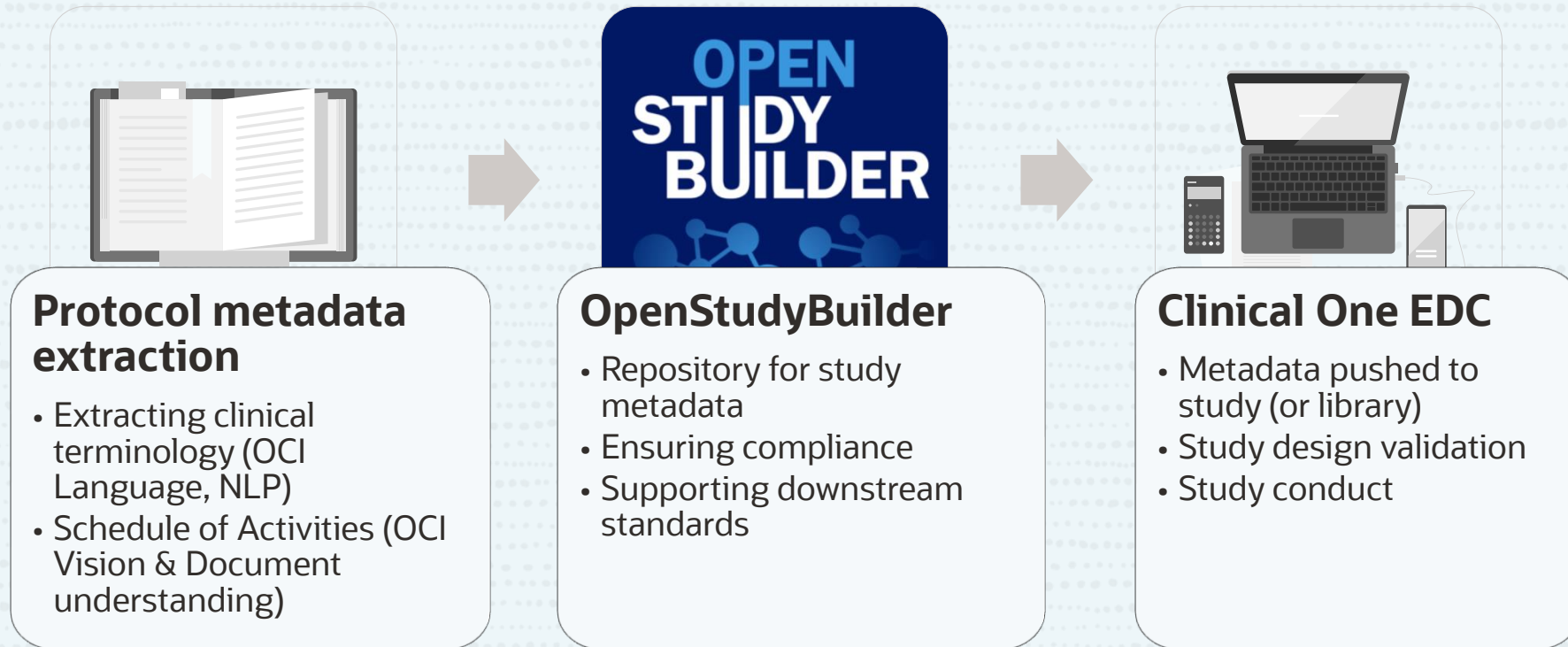
**Automation**

Integration successful!...



... What's next?

# Automation



# Demonstration

---

Metadata Extraction POC for Schedule of Activities

# Schedule of Assessments (SoA) in PDF

STUDY EPOCH	SCREENING	TREATMENT	DOSE ESCALATION		FOLLOW-UP		
	V1	V2	V3	V4	V5	V6	V7
VISIT SHORT NAME	-14	1	29	57	85	91	101
STUDY DAY	±1	±1	±1	±1	±1	±1	±1
VISIT WINDOW (DAYS)	X						
INFORMED CONSENT	X						
DEMOGRAPHY	X						
VITAL SIGNS	X	X	X	X	X		
BODY MEASUREMENTS	X		X	X	X		
MEDICAL HISTORY/CONCOMITANT ILLNESS	X						
PREGNANCY TEST	X	X	X	X	X		
RANDOMISATION		X					
LABORATORY ASSESSMENTS	X	X	X	X			
EPRO		X	X	X			
COMPLIANCE WITH STUDY MEDICATION			X	X	X	X	
END OF STUDY							X
ADVERSE EVENT		X	X	X	X	X	X
CONCOMITANT MEDICATION	X	X	X	X	X	X	X

## OCI Vision Text Detection

Oracle Cloud Vision Text detection interface showing a screenshot of the SoA table. The results panel on the right displays the extracted text from the image, including the table content.



# OCI Document Understanding Table extraction

ORACLE Cloud Cloud Classic Search resources, services, documentation, and Marketplace UK South (London)

Document Understanding

Table extraction  
Identify tables in a document and extract their contents

Document source  
 Demo files  Local files  Object storage

Upload image  
 Drop a file or [select one...](#)  
 Upload image

Output location: Health-ai/OCI4RES34RCH [Select to change output location](#)

Projects  
 Model scope  
 Compartment  
 oraseemeawest (root)

Results *i*

Table results: (1)

Table

VISIT SHORT NAME	V2	V3	VA
STUDY DAY	-14	29	
VISIT WINDOW (DAYS) CONSENT	IF	IF	IF
INFORMED	+		
DEMOGRAPHY	+		
VITAL SIGNS			
BODY MEASUREMENT SIN HISTORY/CO			
MEDICAL			
PREGNANCY TEST	*		+
RANDOMISATION LABORATORY ASS	X	*	+
EPRO	+ I		+
SMEN W COMPLIANCE W		+	
END OF STUD			

STUDY EPOCH

	SCREENING	TREATMENT	DOSE ESCALATION	FOLLOW-UP			
VISIT SHORT NAME	V3	V2	V1	V6	V5	V4	V3
STUDY DAY	13	13	13	57	57	91	117
VISIT WINDOW (DAYS)	13	13	13	57	57	91	117
INFORMED CONSENT	X	X	X	X	X	X	X
DEMOGRAPHY	X	X	X	X	X	X	X
VITAL SIGNS	X	X	X	X	X	X	X
BODY MEASUREMENTS	X	X	X	X	X	X	X
MEDICAL HISTORY/CONCOMITANT ILLNESS	X	X	X	X	X	X	X
PREGNANCY TEST	X	X	X	X	X	X	X
RANDOMISATION	X	X	X	X	X	X	X
LABORATORY ASSESSMENTS	X	X	X	X	X	X	X
EPRO	X	X	X	X	X	X	X
COMPLIANCE WITH STUDY MEDICATION	X	X	X	X	X	X	X
END OF STUDY	X	X	X	X	X	X	X
ADVERSE EVENT	X	X	X	X	X	X	X
CONCOMITANT MEDICATION	X	X	X	X	X	X	X

Images  
 Selected: Screenshot 2024-10-17 at 11.54.56.png



# OCI Data Science Example of scripts

*Output: OSB specific  
JSON format*

ORACLE Cloud COE-NLP-DEMOS Session remaining: 14min Extend ? Sign Out

File Edit View Run Kernel Git Tabs Settings Help

Alpar: X Untitl X activi X finaL X new-v X metac X Health X extrac X 2023-X Launc X Untitl

Code Python 3 (ipykernel)

### OCI Data Science - Useful Tips

- ▼ Check for Public Internet Access
- ▶ Helpful Documentation
- ▶ Typical Cell Imports and Settings for ADS
- ▶ Useful Environment Variables

```
[2]: import requests
response = requests.get("https://oracle.com")
assert response.status_code==200, "Internet connection failed"
```

```
[2]: import pandas as pd

# Example DataFrame simulating the extracted SOA table
data = {
    'STUDY EPOCH': ['Screening', 'Treatment', 'Dose Escalation', 'Follow-up', 'Follow-up'],
    'VISIT SHORT NAME': ['V1', 'V2', 'V3', 'V4', 'V5'],
    'STUDY DAY': [-14, 1, 29, 57, 85],
    'VISIT WINDOW (DAYS)': ['±1', '±1', '±1', '±1', '±1']
}
```

```
[3]: def convert_to_osb_format(row):
# Extract the visit window range (e.g., ±1 becomes min and max values)
visit_window = row['VISIT WINDOW (DAYS)']
min_visit_window_value = -int(visit_window.replace('±', ''))
max_visit_window_value = int(visit_window.replace('±', ''))
# Create the desired format
osb_entry = {
    'epoch_name': row['STUDY EPOCH'],
    'time_value': row['STUDY DAY'],
    'time_unit': 'days',
    'max_visit_window_value': max_visit_window_value,
    'min_visit_window_value': min_visit_window_value,
}
```

Simple 2 \$ 12 Python 3 (ipykernel) | Idle Mode: Edit Ln 6, Col 7 Untitled10.ipynb 0

Video showing interface to upload JSON file, parse it and import SoA parts into OpenStudyBuilder:

*(available from November)*

[https://novo-nordisk.gitlab.io/nnp-public/openstudybuilder/project-description/info\\_integrations/](https://novo-nordisk.gitlab.io/nnp-public/openstudybuilder/project-description/info_integrations/)

# Demonstration



EDC Study Build Automation

# Intelligent Study Build

API integration with Clinical One

## Setup

Standards and study definition

**OpenStudyBuilder**

## Automated build

Metadata transfer  
**Oracle Clinical  
Automation**

Study reveal  
**Clinical One**

## Testing

UAT of study design

**Clinical One**



Video showing setup in OpenStudyBuilder and automated import into EDC Oracle Clinical One

[https://novo-nordisk.gitlab.io/nnp-public/openstudybuilder/project-description/info\\_integrations/](https://novo-nordisk.gitlab.io/nnp-public/openstudybuilder/project-description/info_integrations/)

# Summary

## **OpenStudyBuilder (OSB)**

- Metadata standards across studies and systems
- Consistent collaboration with multiple vendors (best-of-breed)

## **API integration**

- Automation reducing human error

## **Oracle Clinical Automation**

- Simple, easy-to-use interface to manage automation
- Scalability for automating integrations
- Control over automation (push and pull from OSB)

**Better collaboration to drive innovation**

# Moving forward

---



# What's next

## Implementing more **complex study designs**\*

- Branches
- Cycle visits (*using repeating visits?*)
- Randomization schemes
- Drug kits
- Titrations

*\* Dependent on OSB releases*



# What's next

## **Metadata extraction potential applications**

- From protocol to... trial management, data management, data analysis

## **Synthetic data generation**

- Avoid manually entered test data, which is labor-intensive and therefore costly
- Accelerate the trial design process by providing readily available data for validation

- Learn about how Oracle can support this:

<https://blogs.oracle.com/machinelearning/post/announcing-select-ai-for-synthetic-data-generation>

