

Agenda

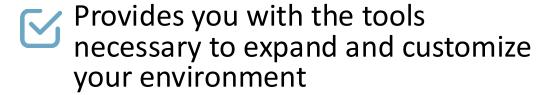
- What is Clarity Studio?
- Administering Objects and Data
- Data Import/Export
- Administering Queries and Portlets
- Administering Lookups

What Is Clarity Studio?





What Is Clarity Studio



In addition to the out-of-the-box investments and objects, Studio allows you to create ancillary components to support Clarity's core features

This components can then allow you to store and query the data in Clarity, and present it in graphical and grid portlets for end-user consumption

Partition Models	
Objects	
Queries	
Portlets	
Portlet Pages	
Menu Manager	
UI Themes	
Views	
Content Add-Ins	
Content Packages	

This section of the Classic UI Administration menu contains the tools that enable an administrator to configure their environment to meet the unique needs of their organization.

What Can Clarity Studio Be Used For?

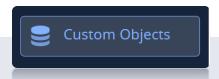
- Modify stock investments or objects to personalize the fields.
- Create objects or sub objects from scratch to supplement other components
- Expose the components via Portlets, Classic UI Views or REST API enable for MUX views, widgets, canvas, etc
- For example:
 - Adding a secondary manager field to Projects
 - Adding Lessons Learned to Status Reports
 - Creating a custom object to track Work Orders
- What have you used Studio for?

Administering Objects

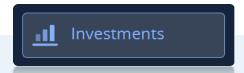




Objects: Choosing the Right Type







Custom Objects are suitable for ...

- Staging tables (integrations)
- Mapping tables
- Large repository for lookup values
- Configuration objects for global controls
- Ancillary, periphery functionality in support of investments

Custom Investments are suitable for ...

- Functionality that has financial component
- Investments that incorporate Risk, Issues,
 Changes, Status Reports, ToDos
- Need financial plans or leveraging transactions
- Investments that will be leveraged in Hierarchies or Roadmap planning
- Leverage a hierarchy of investments to match your process. I.e. An Agile business unit can create Epics, Stories, Tasks, etc.

Objects: Architecting

- Understand limitations of Object Hierarchical structure: Only 3-levels deep (2-levels visible in MUX)
- Use Auto-numbered IDs for user-populated Object/Investment instances for consistent record keeping
- Omit API-enabling objects that are back-end specific and doing require visibility / end-user interaction in the MUX
- Limit the number of attributes per object as this impacts DB views, performance, access rights, among other components
- Choose the right object for a field. Ensuring data is normalized and re-use when possible: e.g., creating a field in the investment

Objects and Attributes: Naming



Use consistent prefixes for Object IDs and Attribute IDs (e.g., your company name for objects ("rego_") and first letter for attributes ("r_")



• Helps identify custom elements vs. out-of-the-box elements

Use consistent style in creating DB and API IDs.

• "p_" prefix for API IDs

- Use underscore separation for DB/API IDs (e.g., r_fy_25_budget)

Use meaningful Object and Attribute IDs that hint at their purpose. X1, X2, X3, X4.....XN are not great names!

Objects: Attribute Administration

When to disable?

- During a transition period when you want to preserve existing data
- When you want to hide / restrict visibility to an attribute in both Classic UI and Modern UX (system wide) (first step to deprecation / retirement)
- When an attribute should not be included in Object XOG-out – this is good way to define which attributes to include in migration to higher environments

When to delete?

- When an attribute is being permanently retired
- When a cleanup is required and data should be purged
- When an object is reaching capacity and the recommended # of attributes is approaching the threshold (i.e., when deletions are necessary to make way for new attributes)

Objects: Auditing Best Practices

Define retention

- Good housekeeping means ensuring that audited data is purged periodically and has a retention period configured
- Ensure that the scheduled purge job is also running

Avoid driving automation from audit data

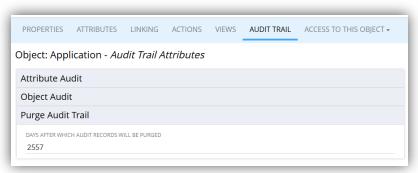
- Audit attributes can easily be modified by unsuspecting admin
- Audit data is volatile, they shouldn't be leveraged for customizations

Be selective

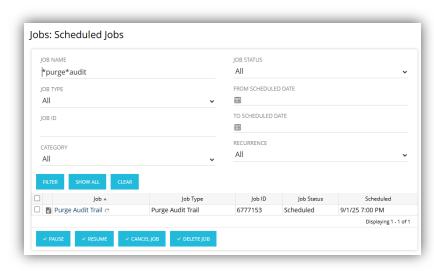
Include only the attributes and audit types required

Be strategic

- Important / meaningful / dependent attributes that hold key information should be prioritized
- Consider what types of changes may be required in future investigations



Application Object with 7-year data retention



Recurring "Purge Audit Trail" scheduled job

Administering Queries & Portlets





NSQL Queries and Portlets: Best Practices

When working with queries (Processes, Lookups, and/or Portlets):

- Understand governors and applications limits
- Consider where the query is placed, its logic and how often it will run
- Consider access to Portlets, Portlet Pages and MUX Pages (if portlets are channeled). Ensure only required columns are used in the query.
- ☑ Consider performance implications to using Security construct in NSQL queries to security data: @WHERE:SECURITY:<entity type>:<entity id>@

NSQL Queries and Portlets: Best Practices

When working with queries (Processes, Lookups, and/or Portlets):

- - -BROWSE-ONLY
 - -SECURITY @WHERE:SECURITY:ISSUE:i.odf_pk@, @WHERE:SECURITY:PROJECT:inv.id@
 - -@WHERE:PARAM:LANGUAGE@, @WHERE:PARAM:LOCALE@
 - -@UPPER@, @SYSDATE@, @NVL@, @SUBSTR@, @DBUSER@, @+@
 - -Functions COP_DATE_TRUNC_FCT
 - -@WHERE:PARAM:USER_ID@
 - -COALESCE

NSQL Queries and Portlets: Best Practices

When creating a portlet:

- Consider volume of data and how often a query will run as it can impact the application.
- Choose the right portlet type, leverage charts where applicable to create a rich experience.
- Ensure User Access is properly setup before publishing.

Administering Lookups





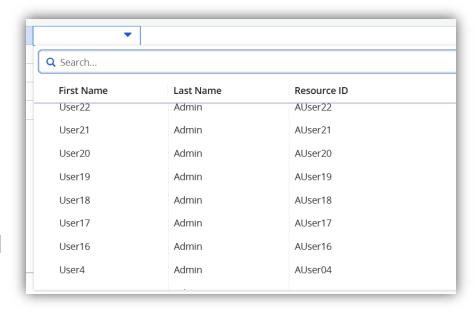
Lookups: Best Practices

LOOKUP VALUES: Code vs. Enum

- Enum allows for future realignment/expansion
- Can re-label "Name" without impacting underlying value meaning in DB
- Avoid using spacing in lookup IDs (code)

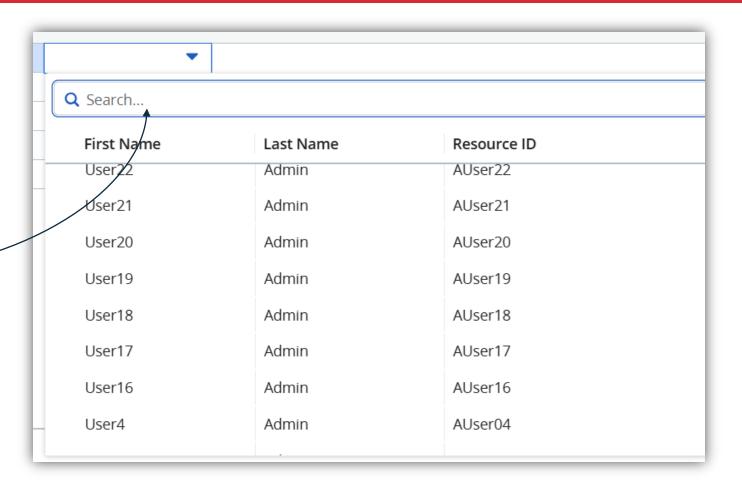
LOOKUP TYPES: Static vs. Dynamic

- When to use **Static**:
 - Simple list of values
 - No security constraints on LOVs
 - Best to limit size of static lists (long page loading times when modifying large lists)
 - Leverage custom object over custom lookup when LOVs exceed reasonable size (~100)



Lookups: Best Practices

- When to use **Dynamic**:
 - More complex logic required
 - LOVs have dependencies against data in adjacent fields (parameterized)
 - Need expanded autosuggest values (multicolumn hints in drop-down select)
 - LOVs have access-based dependencies (@WHERE:PARAM:SECURIT Y@)



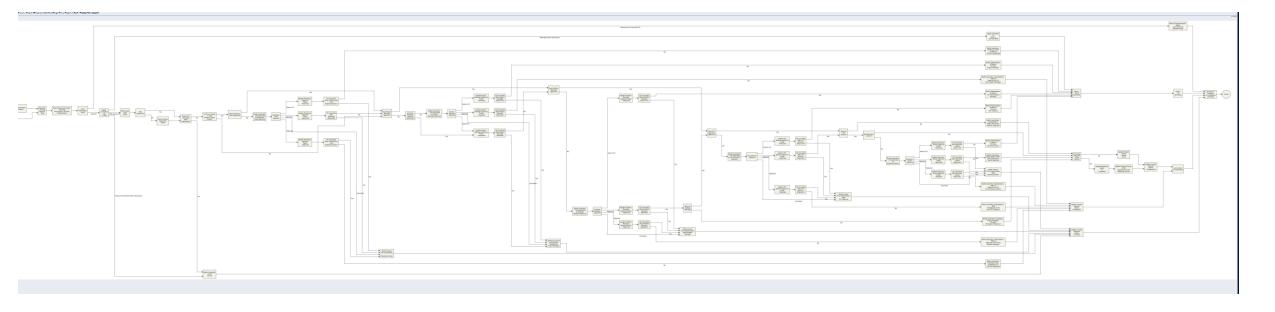
Processes



regoUniversity

Processes: Best Practices

Friends don't let friends create processes like this:



Processes: Best Practices

- Processes are a key component in Clarity to leverage automations and process data:
 - Avoid the need to enter data in multiple places
 - Allow for automations that align with your business process i.e. Project Stage Gates, Timesheet approvals, etc.
 - Really powerful, allowing custom scripting, interacting directly with the database and even creating integrations
- Keep processes as simple as possible, they are intended to be short lived
- Avoid complex processes that involve long running logic or a high number of steps as it represents a load to the BG service.
- Sub processes Use with extreme caution
- Limit the number of processes that might be triggered at the same time
- Be careful with what jobs you call from processes

Processes: GEL Standards

- GEL is one of the most powerful components to customize the tool. Allowing for custom code to be ran within processes.
- They should be treated like any other programming language / development:
 - Ensure the code is easy to read, well-structured and formatted
 - Code should be commented, providing meaningful details on its version, purpose and the overall logic
 - Make sure proper error handling is implemented and handle accordingly for the script to terminate gracefully.
 - Be mindful when logging messages and avoid excessive logging. Using a debug flag can help during development
 - Remove dead code / cleanup when applicable
 - When interacting with the database, assess the risk and avoid impacting data integrity
 - Embrace REST API and XOG

Considerations and Recommendations





Other Studio Features

- Views
 - Allow Object views in Classic UI to be customized. Admins can build pages, sections and attribute layouts along with actions for ease of use.
- Menus and Links
 - Allows Admins to customize the Classic UI menu for a friendlier navigation experience
- UI Themes
 - Allows Admins to use custom CSS to personalize the Classic UI

Modern UX

- Modern UX (a.k.a. MUX) is the latest UI, providing a more reactive experience and being heavily invested into by Broadcom.
- Provides the latest functionality for:
 - Views, Grids and Blueprints
 - Leverages automations through the UI, allowing admins to create: Business Rules, Notifications and Action Items
 - Provides a reach UX experience leveraging: powerful Grid allowing per user views, Widgets, Canvas and the new reporting module.
- All best practices still apply:
 - MUX relies on Objects, Attributes and Lookups
 - CITs will be exposed via the MUX and admins need to be mindful when deciding between an object or a CIT
 - A high volume of views, Business Rules, Notifications or complex/heavy Canvas dashboards will impact the user experience and can cause performance issues in the application.

Additional Considerations

- Ensure proper documentation is kept for all Clarity customizations and tracking.
- Create a support process around the Clarity admin activities:
 - Properly test and plan for rollout of any customizations
 - During deployments, ensure they are done when the environment has low concurrency and clear cache to ensure it properly repliaces
 - Housekeeping, Housekeeping! Review, cleanup and monitor your customizations and volume of data:
 - In a scheduled basis
 - When assessing/designing new customizations
 - When rolling out new customizations

Want to know more? Join our Housekeeping class!

Questions?



regoUniversity
2025



Master Clarity with Rego University

Earn Certifications in Administration, Leadership, and Technical Proficiency

Let Rego be your guide.



Elevate Your Professional Expertise with Rego University Certifications

Rego is excited to continue our **certification programs**, designed to enhance your expertise in Clarity administration, leadership, and technical skills. These certifications provide hands-on experience and knowledge to excel in your career.









Certification Requirements:

✓ Completion: 12 units per certification track

Eligibility: Open to all Rego University attendees

Important Reminder:

To have your certification **credits tracked**, ensure you **complete the class surveys in the app** after each session. This step is critical for certification progress.

Please take a few moments to fill out the class survey. Your feedback is extremely important for future events.



Thank You for Attending Rego University

Instructions for PMI credits

- Access your account at pmi.org
- Click on Certifications
- Click on Maintain My Certification
- Click on Visit CCR's button under the Report PDU's
- Click on Report PDU's
- Click on Course or Training
- Class Provider = Rego Consulting
- Class Name = Rego University
- Course **Description**
- Date Started = Today's Date
- Date Completed = Today's Date
- Hours Completed = 1 PDU per hour of class time
- Training classes = Technical
- Click on I agree and Submit



Let us know how we can improve! Don't forget to fill out the class survey.



Phone

888.813.0444



Email

info@regoconsulting.com



Website

www.regouniversity.com

Continue to Get Resources and Stay Connected

- Use <u>RegoXchange.com</u> for instructions and how-tos.
- Talk with your account managers and your Rego consultants.
- Connect with each other and Clarity experts at RegoGroups.com.
- Sign up for webinars and join in-person Rego groups near you through at Rego groups
- Join us for the next Rego University!

