

# ★ LAS VEGAS ★

## regoUniversity

### 2025

Sponsored by

**ValueOps®**  
by Broadcom

**Clarity®**  
by Broadcom

**Rally®**  
by Broadcom

**ConnectALL**  
by Broadcom

**Insights**  
by Broadcom

# Security Models and OBS

Your Guides:  
James Gille and Chris Ciavarella

Gold Sponsor



# Introductions

- Where do you work?
- What's your role?
- What area/topic would you like to learn more about?
- Anyone still exclusively on the Classic UI?





# Agenda

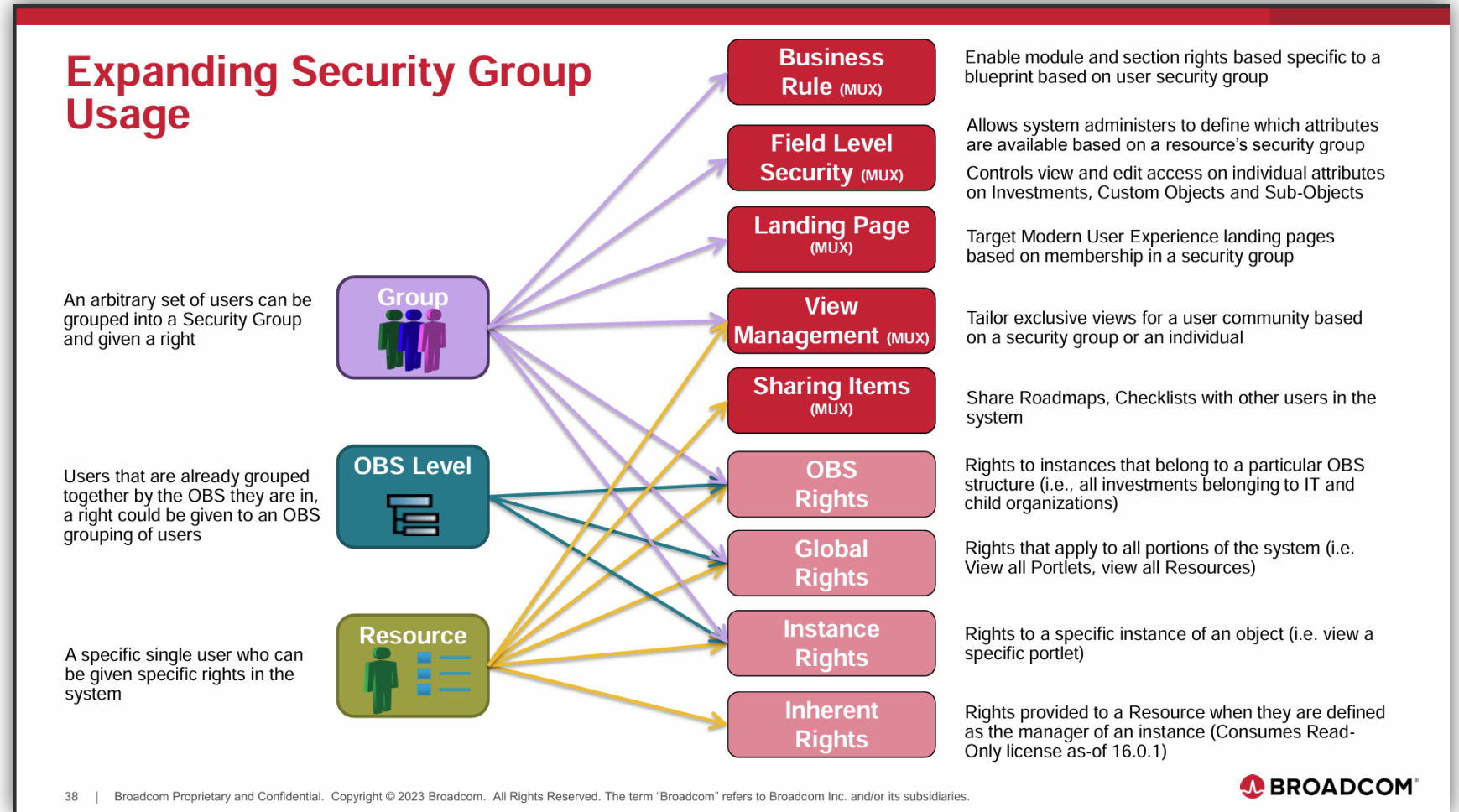
---

- Introduction to Security
- Security Rights
  - Global Rights
  - OBS Rights
  - Instance Rights
  - Inherent/Automatic Rights
  - Psuedo-Security: Blueprint Rules
- Organizational Breakdown Structures (OBS)
  - OBS Demo / Live Session
- Security Best Practices

# Introduction to Security

# Complex Security & Rego's Value

- Although a bit-outdated, this Broadcom slide does well to articulate the complex, inter-related nature of security within Clarity.
- Our recommendations in this space are particularly key due to that complexity and potential to impact performance and administrative overhead.



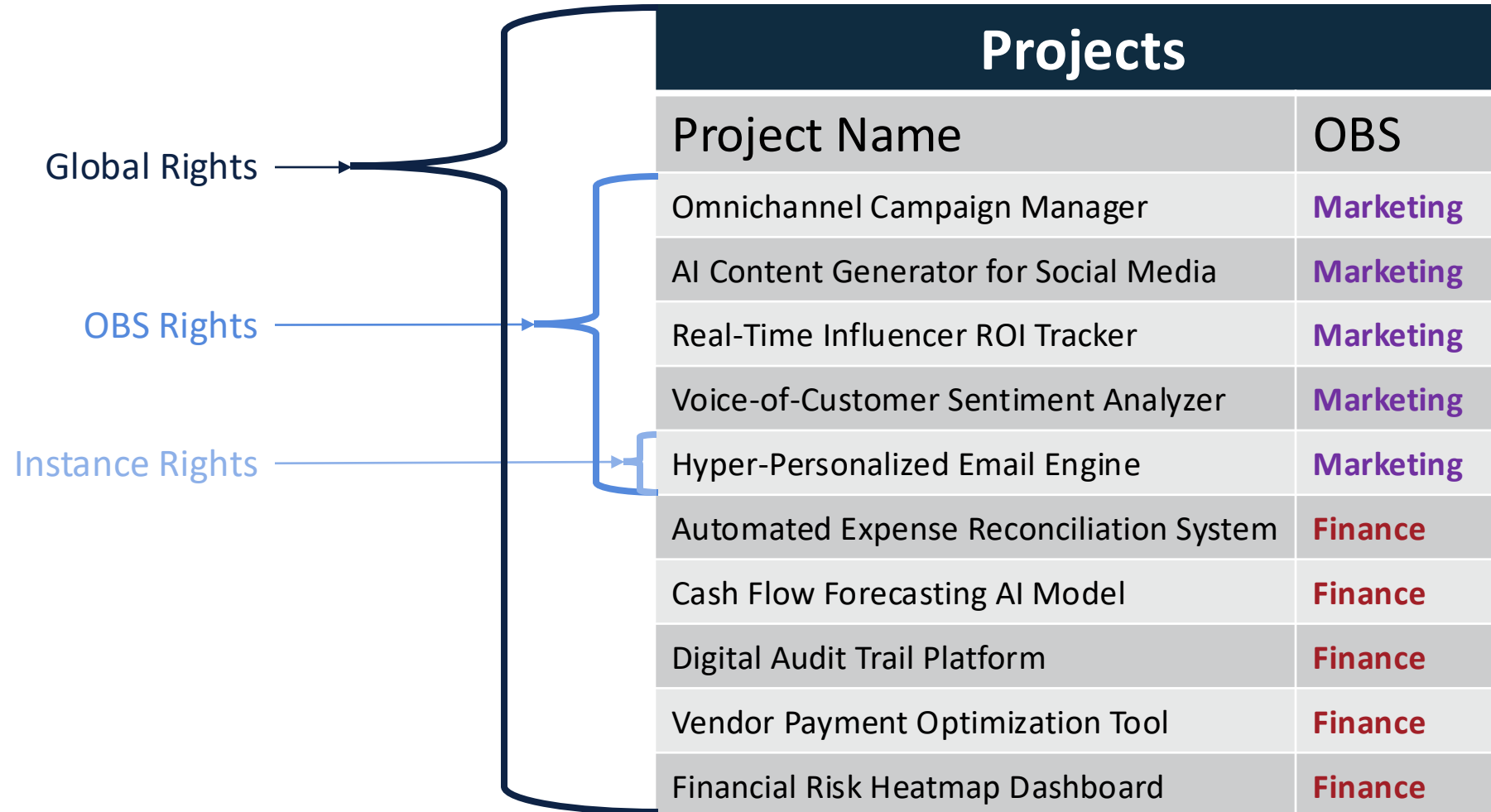
# Security Considerations

- Types of rights
  - Global – all records of a given type (all projects, all resources, etc.)
  - OBS – all records of a given type associated with a specific OBS
  - Instance – one specific record (a single project, resource, etc.)
  - Inherent – automatically assigned by Clarity
  - Field-Level Security (FLS) – Modern UX enables view and edit security at individual attribute level
- Can be assigned at group or individual level
  - Best practice is group level
  - Define groups based on needed functionality (e.g., PM Group, RM Group, Admin group)
  - Provides option to restrict access to menu options, pages, etc.
  - Provides option to secure sub-objects and sub-pages (Classic PPM)
  - Provides option to notify everybody in a group (e.g., Results of Job execution)
- NOTE – the rights you grant can impact your licensing costs!



# Security Rights

# Security Rights – Overview





# Security Rights – Global Rights

- Global Rights have the broadest scope
- Overrides any other associated instance or OBS level rights
- Provides broad access to objects of a particular type
- Allows users access a general area of the application to perform a specific function or to all instances of an object
- A common global right is Resource – View All
- Recommended where possible for performance reasons

PROPERTIES RESOURCES GROUP'S ACCESS RIGHTS ▾		
Group: RM Finance - <i>Global Access Rights</i>		
<input type="checkbox"/>	Access Right ▲	Description
<input type="checkbox"/>	Idea - Cost Plan - View All	Allows resource to view all the Idea cost plans.
<input type="checkbox"/>	Idea - View - All	Allows user to view all Ideas.
<input type="checkbox"/>	Idea - View Financial Information All	Allows resources to view financial properties of all ideas.
<input type="checkbox"/>	Idea Management - Navigate	Allows user access to the idea management pages in the new user experience. The user will only be able to view information for ideas to which the user has access.
<input type="checkbox"/>	Menu Links - Navigate	Allows user to access the Menu Links page in the new user experience.
<input type="checkbox"/>	Menu Links - View - All	Allows user to view all sections and links in new user experience.
<input type="checkbox"/>	Posted Transactions - View All	Allows user to view all projects, ideas and custom investments posted transactions.
<input type="checkbox"/>	Project - Budget Plan - View All	Allows resource to view all the Project budget plans.
<input type="checkbox"/>	Project - Cost Plan - View All	Allows resource to view all the project cost plans.
<input type="checkbox"/>	Project - Risk, Issue, Change Request - View - All	Allows user to view all risks, issues, and change requests for a project instance.
<input type="checkbox"/>	Project - View Financial - All	Allows resources to view financial properties of all projects.
<input type="checkbox"/>	Project - View Management - All	Allows user to view the general and management properties and processes on all Projects which have been enabled for management.
<input type="checkbox"/>	Project - View Tasks - All	Allows user to view the tasks and work breakdown structure for all Projects to which the user has access.
<input type="checkbox"/>	Project Management - Navigate	Allows user access to the project management pages in the new user experience. The user will only be able to view information for projects to which the user has access.
<input type="checkbox"/>	Resource - Navigate	Allows user access to the Resource list and Resource Finder pages. The user will only be able to view resource information for resources to which the user has access.
<input type="checkbox"/>	Resource - View - All	Allows user to view all resources and their information except the financial properties of the resource. This right is dependent on Resource - Navigate being granted.
<input type="checkbox"/>	Resource - View Book - All	Allows user to view bookings for all resources.

*Global Rights will provide access to “All” records of a given type, as denoted in the Access Right name and description*

# Security Rights – OBS-Level Rights

- OBS-Level Rights define access to a logical grouping of items
  - Investments grouped by Department
- More limited access based on the OBS that a record is associated with
  - e.g., Project – View access to all projects where Department OBS = Corporate/Marketing
- Common use case is to isolate resources or projects for discrete organizations

Organizational Breakdown Structures

<input type="checkbox"/>	OBS ▲	Used for Access Rights
<input type="checkbox"/>	Business Location	✓
	Corporate Department OBS	
	Corporate Location OBS	
<input type="checkbox"/>	Team OBS	✓

NEW   ✓ DELETE   ✓ USE FOR ACCESS RIGHTS   ✓ REMOVE USE FOR ACCESS RIGHTS

If an OBS will be used to define access to a logical grouping of instances, ensure that it is selected for **Use for Access Rights**.

***\*Note:** When an OBS is **not** used for access rights, its purpose is more suited for tagging (organizing data into logical groups) and then reporting/filtering on those selections only.*

# Security Rights – Instance Rights

- Provide access to a specific instance of an object (one project, one resource, etc.)
- Can be assigned to an individual or a group
- Can lead to performance issues if not managed carefully
- ***Difficult to manage*** - minimal use is recommended

Resource: Clarity Administrator - *Instances for This Access Right*

INSTANCE RIGHT  
Status Report - Edit

DESCRIPTION  
Allows resource to edit specific Status Report objects.

> Filter: System Default ▾

<input type="checkbox"/>	Overall Status	Status Report Name	Report Date ▾	Report Status	Schedule Status	Scope Status	Cost and Effort Status
<input type="checkbox"/>	◆	Status Report	4/18/18	Final	◆	◆	◆
<input type="checkbox"/>	⚠	Status Report	4/15/18	Draft	⚠	◆	◆
<input type="checkbox"/>	⚠	Status Report	4/11/18	Final	✖	◆	✖
<input type="checkbox"/>	◆	Status Report	4/11/18	Final	◆	◆	◆
<input type="checkbox"/>	⚠	Status Report	4/8/18	Final	⚠	⚠	◆
<input type="checkbox"/>	◆	Status Report	4/8/18	Final	◆	◆	◆
<input type="checkbox"/>	⚠	Status Report	4/4/18	Final	⚠	◆	◆
<input type="checkbox"/>	⚠	Status Report	4/4/18	Final	⚠	◆	◆
<input type="checkbox"/>	◆	Status Report	4/1/18	Final	◆	◆	◆
<input type="checkbox"/>	⚠	Status Report	4/1/18	Final	⚠	⚠	⚠
<input type="checkbox"/>	⚠	Status Report	3/22/18	Final	◆	⚠	◆
<input type="checkbox"/>	⚠	Status Report	3/22/18	Final	⚠	⚠	⚠
<input type="checkbox"/>	◆	Status Report	3/20/18	Final	◆	◆	◆
<input type="checkbox"/>	◆	Status Report	3/15/18	Final	◆	◆	◆



# Security Rights – Inherent/Automatic Rights

- Automatically applied when user is first added to Clarity or the user is associated with a particular field on an object (e.g., Resource Manager)
- **Irrevocable**
  - User Favorites Menu – Edit
  - Project – Manager (Auto)
  - Resource – Manager (Auto)
- **Revocable**
  - Resource – Enter Time (for themselves)

PROPERTIES   OBS AND PARTITIONS   GROUPS   **RESOURCE'S ACCESS RIGHTS**

Resource: Chris Ciavarella - *Instance Access Rights*


Below are the rights this resource has for specific object instances.

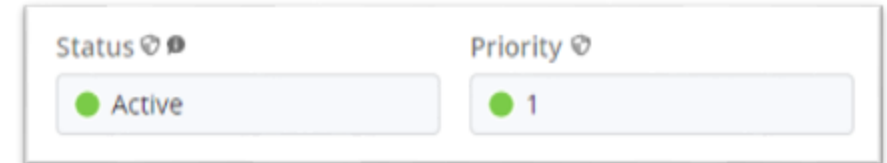
ACCESS RIGHT  
\*auto\*

OBJECT  
All

<input type="checkbox"/>	Access Right ▲	Object
<input type="checkbox"/>	Hierarchy - Creator (Auto)	Hierarchy
<input type="checkbox"/>	Resource - Manager (Auto)	Resource
<input type="checkbox"/>	Resource - Self (Auto)	Resource

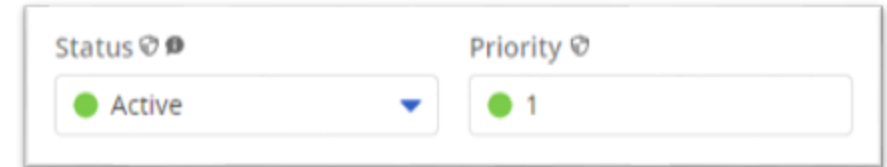
# Security Rights – Field-Level Security (FLS)

- Controls access to a specific attribute of an object
- Available in Modern UX under:  
**Administration -> Attributes**
- Can grant view and edit rights or hide a field entirely
  - \*When inaccessible secured fields comprise an entire blueprint section, that section is not rendered.
- Can lead to performance issues if not managed carefully
- Is denoted by a "shield" icon in the field label 
- **Difficult to manage** - minimal use is recommended



The screenshot shows two input fields: 'Status' and 'Priority'. The 'Status' field contains the text 'Active' and the 'Priority' field contains the number '1'. Both fields are greyed-out, indicating they are read-only for the current user.

*When a secured field is read-only for the current logged-in user, it appears greyed-out.*

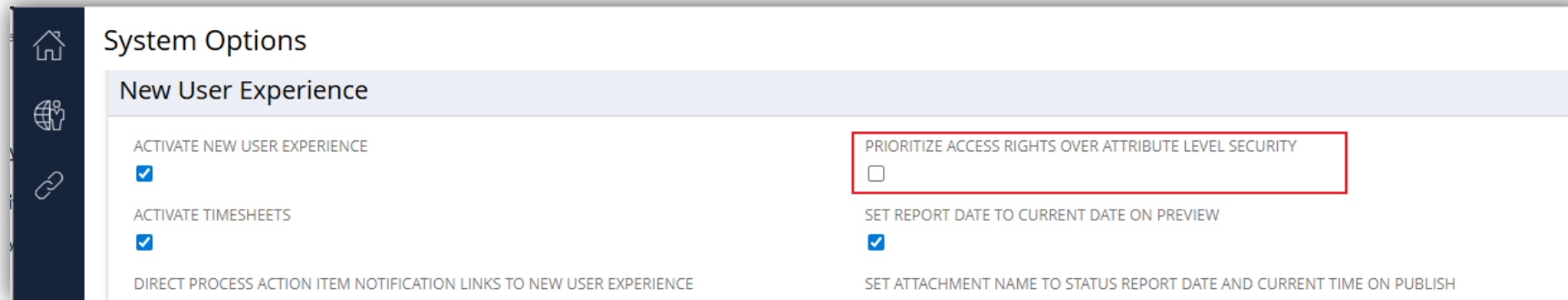


The screenshot shows two input fields: 'Status' and 'Priority'. The 'Status' field contains the text 'Active' and the 'Priority' field contains the number '1'. Both fields are active and editable. A small shield icon is visible in the 'Status' field label, indicating it is secured.

*When a secured field is editable for the current logged-in user, it renders normally but the "shield" is still denoted in the label*

**Best practice:** Prioritize Group and OBS access rights over Field-Level Security

# Security Rights – Field-Level Security (FLS)



This system option under **Administration** -> **System Options (Classic PPM)** controls the behavior of FLS.

## Case Study - Resource: James

*Global Rights:* James has **view-only** rights to a project.

*FLS Rights:* James is a member of a group that has **view** and **edit** rights to the project "Manager" field.

### Scenario A (default):

Prioritize Access Rights is **disabled**.

**Result:** James **can** edit "Manager" field.

### Scenario B:

Prioritize Access Rights is **enabled**.

**Result:** James **can not** edit "Manager" field.



# Pseudo-Security: Blueprint Rules

- Records (instances), attributes, sections and modules (tabs) of a Blueprint in the MUX can be conditionally “secured” or hidden using Blueprint-based Business Rules
- This type of security can still be circumvented by Classic PPM, XOG, REST APIs or Processes
- Read-Only Blueprint rules always take precedence over Field-Level Security ‘Edit’ access

The screenshot shows the 'View Rule' configuration window. It includes fields for 'Rule Name' (set to 'Hide Section until Approved') and 'Description' (placeholder: 'Enter the rule description'). The 'Target Object' is set to 'Project'. Under 'Rule Type', 'View Page' is selected. The 'Conditions' section shows 'Always True' selected. The 'Match Filters' section shows a filter for 'Status' with the value '≠ Approved'. The 'Actions' section shows a list of actions: 'Hide Sections', 'Settings', 'Stakeholders', and 'Project Summary', with a '+ Select' button and a '+ Add actions' button at the bottom.

# Blueprint Rules vs. FLS

Individual attributes in Clarity can be conditionally secured or hidden using either Blueprint-based Business Rules or using FLS (field level security) at the attribute-level. Some initial considerations to help determine the optimal approach given the requirements:

Feature \ Requirement	Business Rule	Field Level Security
Supported Number of Attributes	Users can define any number of rules but only 50 rules can be active simultaneously.	No restriction
Supported Attribute Types	Cannot be used to secure time-scaled attributes.	A minimal number of attributes are flagged as "Not Securable."
Performance	Not formally evaluated	Not formally evaluated
Visual Indicator	Field is "greyed out" with hover message that indicated field is locked by a business rule	Field is "greyed out" with shield icon visible to the right of the attribute label
Securing Logic	Can be aligned to both group membership and other attribute conditions and triggers	Relies strictly on group membership
Real-Time Application	May require a screen refresh if business rules is triggered based on real-time condition being met	Always active
User Experience	Applies to Modern UX only.	
Evaluation Priority	Most restrictive takes priority.	
Priority Against Security Rights	Most restrictive takes priority.	FLS takes priority over security rights, unless "Prioritize Access Rights over Attribute Level Security" is selected in system options.

# Organizational Breakdown Structures (OBS)



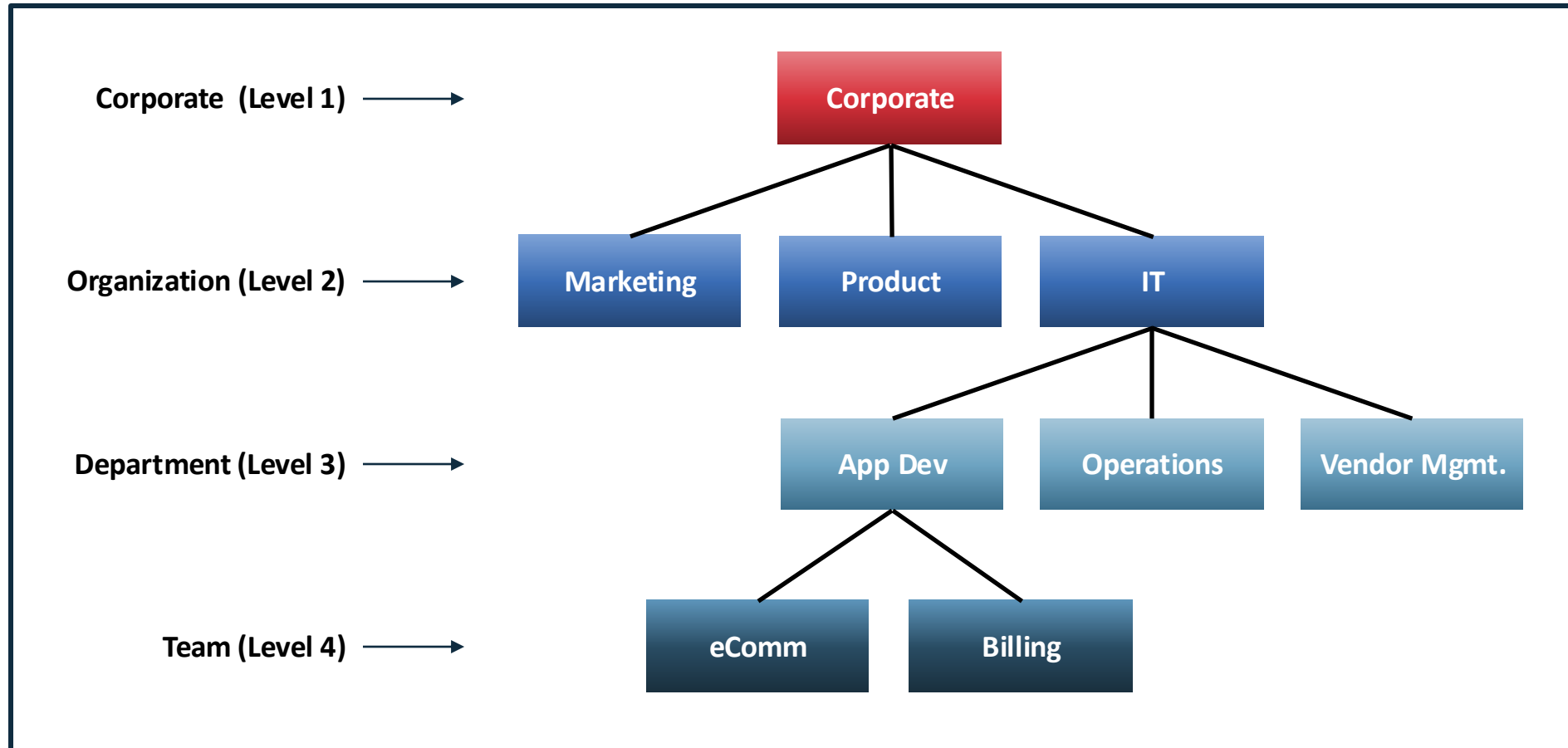
# Organizational Breakdown Structure (OBS)

- An OBS is a company-defined hierarchical structure
- Can be used for filtering, reporting, and security (OBS rights)
- You can create any number of OBSs with up to 10 levels
- Security can be enabled for any OBS
  - Department OBS and Location OBS only on initial creation (until tied to Entity)
- Remember that every OBS created will need to be maintained which can be a time-consuming task depending on complexity
- OBS is a default filter parameter on most out-of-the-box reports

# OBS Types

- Identifying what your business needs are is the first step in determining what types of OBS structures to build
  - In what ways do we need to group data for portlets and reports?
  - Are there other fields that can be used to accomplish this? (e.g., Sponsoring BU)
  - Is there a need for multi-level filtering or roll-up reporting?
- Some OBS Examples:
  - Organizational (like HR org chart)
  - Financial Departments / Cost Centers
  - Functional Organizations and Teams
  - Product or Product Teams

# OBS Example: Functional / Team OBS



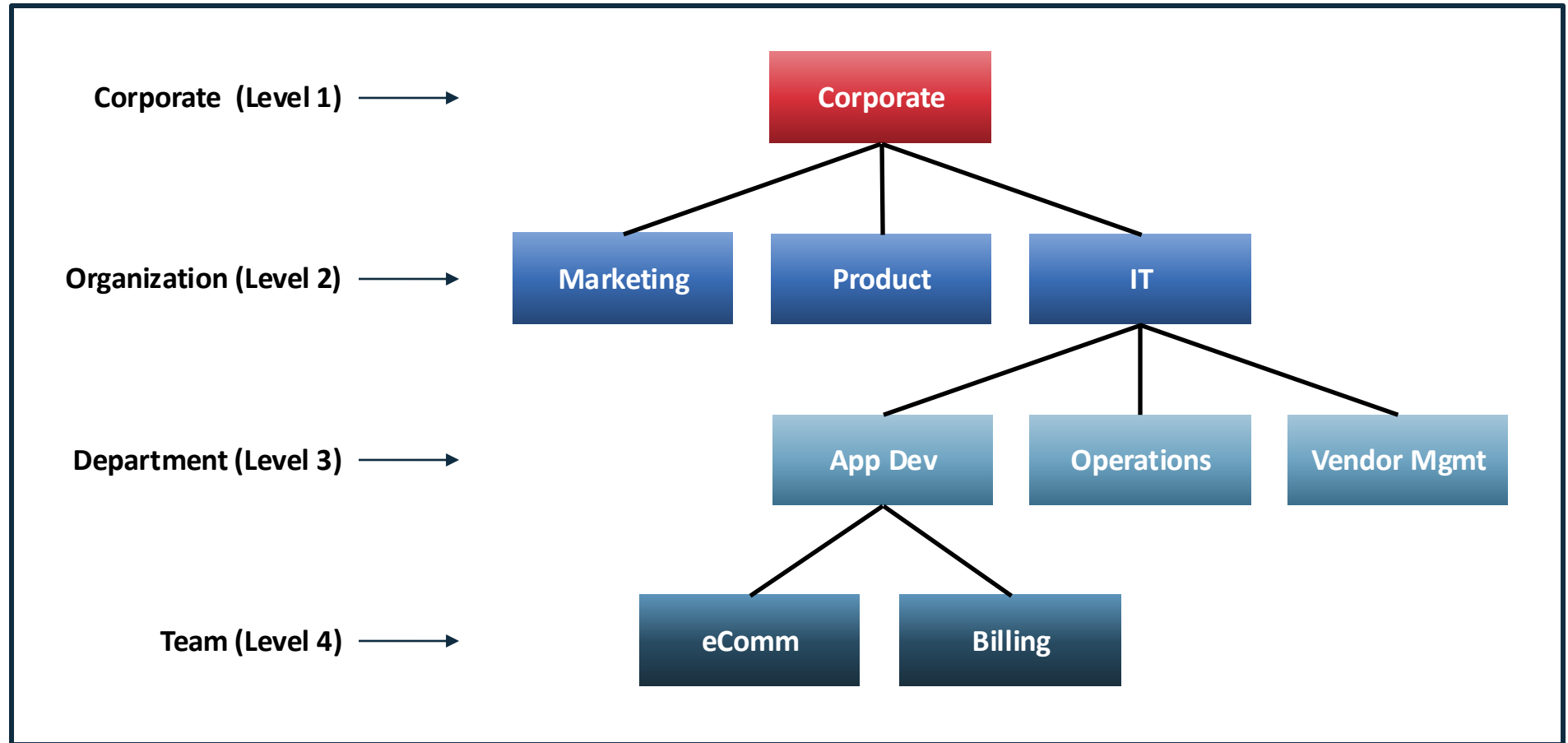


- Maintaining an OBS and associated records can be time-consuming
- Keep the number and complexity of OBSs to a minimum
- Changing the structure and/or labels in an OBS can affect reporting
- Deleting an OBS unit will also delete all child units under that level
- Any record tied to a deleted OBS unit will be “orphaned” (null value in OBS field)
- Units can be moved from one parent to another within a specific OBS and all child units will follow
- OBS associations can be done using a list/grid view, but the fastest method could be from the Admin menu (OBS Unit tab “Attached Instances”)
- Look for automation opportunities (e.g., do associations via GEL script)

# OBS Demo: Let's Build this OBS!

## Steps:

1. Create the OBS
2. Enable Security
3. Define the Levels
4. Add Units
5. Attach Instances



# Security Best Practices

# Security Best Practices

- Assign people their rights only once
  - Granting the same rights at different levels or through multiple groups can slow down performance in some pages/views
- Only use Instance rights as a last resort and try to keep at a Group level
- Minimize security maintenance
  - It's better to create more Groups with a relatively small number of rights than to create a few Groups with rights to many different objects.
  - This approach makes it easier to add and remove rights when people change roles, you can simply add them or remove them from the appropriate groups
- Understand what each right means
  - Know the definition of each right
  - Understand the licensing implications of different rights



# Security Best Practices

- Think about security “roles” or user “personas”
  - Not the same as Primary Role
  - Theoretical, not in the system (examples: Timesheet user, Project approver)
- Document your security model design before building it
  - Use OOTB groups as a reference but most likely you will create your own groups
- Set up the model in Clarity
  - Create dummy data
  - Create generic test users – one per functional “role”
- Use the MUX’s “Impersonate” feature to verify the model is working as designed
  - Can you see and do what you expect for each role?
  - Can you see or do anything that shouldn’t be allowed?

# Security and Group Design

There tends to be two general design philosophies, one organized around personas and one around capabilities.

## Persona

- One "function" group established for each persona (e.g., Project Manager, Portfolio Manager). This group establishes access to capabilities.
- Multiple "data" groups are established for each persona that grants access to data records, usually driven by OBS.
- Blueprints, Blueprint Rules, and FLS are used to support department, team, or organizational variations.
- Ideally, a given user is associated with only one persona group for simplified provisioning.
- This model often requires that security rights are included in multiple groups. For example, a project view right may need to be included in the PM, PfM, RM, and Executive persona groups. Maintenance is the tradeoff here.
- A variation in which personas are stacked (i.e., a PfM must be a member of the Team Member and PM groups) can help rationalize rights, but results in overly complicated provisioning.

## Capability

- Security groups are aligned to high-level capabilities (e.g., Idea Management, Schedule Management).
- Users are provisioned based on what they need to do in the system, with each user potentially have a different set of groups.
- This model reduces maintenance overhead, but provisioning can be complicated.
- The model can also leverage the "data" group concept, though they tend to be more broadly defined than when established for individual personas.

# Security and Group Design

---

Establish and communicate "Guiding Principles" during initial design sessions:

- Lean towards an open model, locking down data only when required.
- Don't assume malicious intent. Security is enabled to support the user experience and meet visibility, compliance, or regulatory requirements.
- Configure auditing as an enabler to an open security model, mitigating the risk of accidental updates.
- Favor Global rights over OBS and favor OBS over Instance.
- Continuously look for opportunities to automate provisioning (e.g., expand inherent rights).
- Limit the number of security groups per resource. Minimize overlapping rights; try to have resources receive their rights from one source only.

# Security Groups

- Groups

- A set of users who perform similar functions or roles in the tool
- Contains a collection or combination of rights applicable to each member of that group
- Can associate a single resource with any number of groups
- Can contain a collection of Instance, OBS, and Global level rights
- Quickest way to assign multiple rights to several people at once
- Much easier to manage than assigning rights to specific individuals

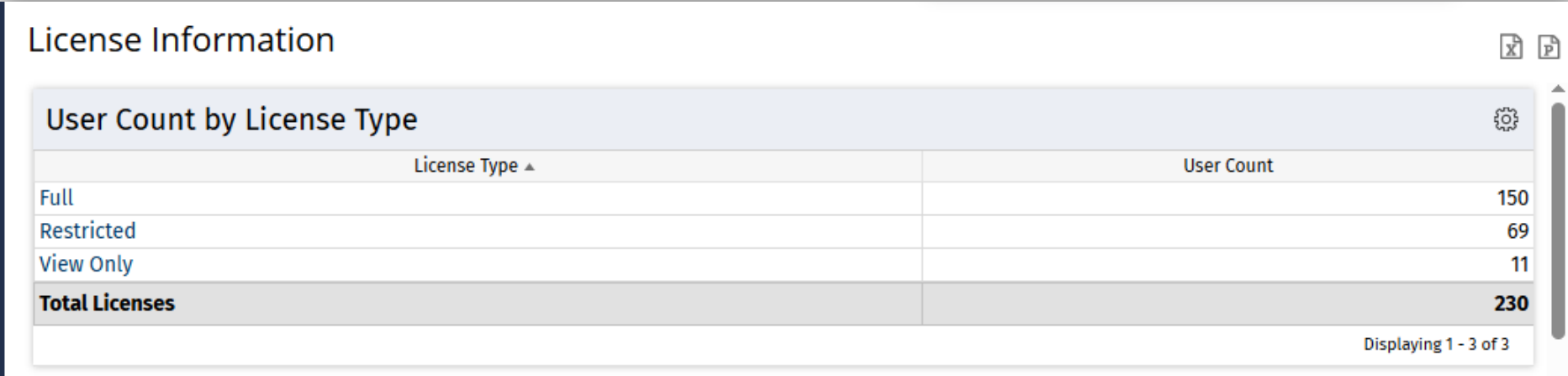
- Other Uses for Groups

- Can be used to determine what users see in the application
  - Menu options
  - Pages, Portlets and Tabs
- Groups can also be created for things other than granting rights
  - e.g., notifications from a workflow



# Security and Licenses

- The rights assigned to an individual or group determine the type of license consumed and counted by Clarity
- Use the License Information portlet (Administration->License Information) to monitor usage and maintain compliance
- Audit this usage periodically



User Count by License Type	
License Type ▲	User Count
Full	150
Restricted	69
View Only	11
<b>Total Licenses</b>	<b>230</b>

Displaying 1 - 3 of 3

# Questions?







# 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.



# Surveys

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 = **RegoUniversity**
- 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](mailto:info@regoconsulting.com)



### Website

[www.regouniversity.com](http://www.regouniversity.com)

# Continue to Get Resources and Stay Connected

- 1 Use [RegoXchange.com](https://RegoXchange.com) for instructions and how-tos.
- 2 Talk with your account managers and your Rego consultants.
- 3 Connect with each other and Clarity experts at [RegoGroups.com](https://RegoGroups.com).
- 4 Sign up for webinars and join in-person Rego groups near you through at [RegoConsulting.com](https://RegoConsulting.com)
- 5 Join us for the next [Rego University](#)!

