



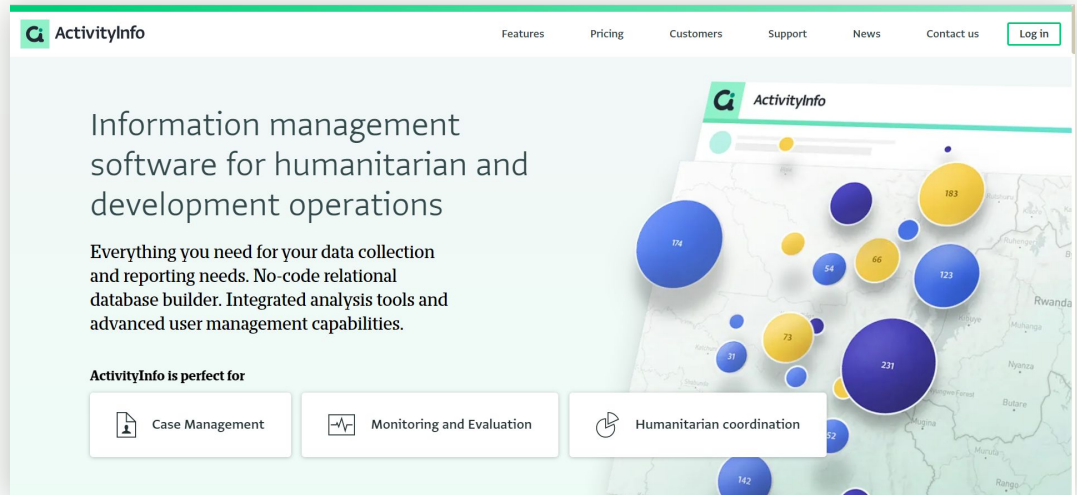
ActivityInfo

Data protection in practice - Best practices for designing Roles in ActivityInfo

Starting shortly, Please wait!

Presented by the ActivityInfo Team

- Track activities, outcomes
- Beneficiary management
- Surveys
- Work offline/online



The screenshot shows the ActivityInfo website homepage. At the top, there is a navigation bar with the ActivityInfo logo on the left and links for Features, Pricing, Customers, Support, News, Contact us, and a Log in button on the right. The main content area features a large heading: "Information management software for humanitarian and development operations". Below this, a sub-heading reads: "Everything you need for your data collection and reporting needs. No-code relational database builder. Integrated analysis tools and advanced user management capabilities." Underneath, a section titled "ActivityInfo is perfect for" contains three icons with corresponding text: a document icon for "Case Management", a line graph icon for "Monitoring and Evaluation", and a circular arrow icon for "Humanitarian coordination". On the right side of the page, there is a map of Rwanda with several colored bubbles (blue and yellow) of varying sizes, each containing a numerical value representing data points.

Meet your instructors



Jeric Kison

Customer Success Director
BeDataDriven



Victoria Manya

Customer Education Specialist
BeDataDriven

Data Security Webinar Series

Cybersecurity Awareness Month

OCT 5

SESSION 1

Top 5 data security risks for M&E professionals and what you can do about them

OCT 12

SESSION 2

Data protection in practice - Best practices for designing Roles in ActivityInfo

OCT 19

SESSION 3

Office Hours - Designing Roles in ActivityInfo

Outline

- 1 Data security principles
- 2 Understanding roles in ActivityInfo
- 3 How to create roles in ActivityInfo
- 4 Best practices for designing roles
- 5 Q&A

Poll

What challenges have you faced when managing Roles and Permissions in your organization's database?

- **Complexity:** Dealing with intricate permission setups.
- **Role Overload:** Managing too many roles and tailoring roles to specific needs.
- **Security Balance:** Balancing data protection and access.
- **Onboarding Time:** Time-consuming user setup and Clarifying roles with users.
- **Testing Burden:** Rigorous testing before implementation.
- **Audit Tracking:** Keeping records of changes.
- Resolving **overlapping** permissions.
- **Scalability** Challenges: Adapting to expansion.
- **Documentation gaps:** Keeping role documentation current and Lack of user training.

1

Data security principles

What is data security?



Confidentiality

Confidential data protected from exposure to unauthorized parties



Integrity

Prevention of unauthorized changes to your data



Availability

Ensuring data is available when needed to the parties who need it

Principle of least privilege

*“a security architecture should be designed so that each entity is granted the **minimum** system resources and authorizations that the entity needs to **perform its function**”*

- US Committee on National Security Systems

What role does the M&E professional play in data security?

- Planning M&E systems
- Planning data collection
- Sharing data with internal and external stakeholders
- Communicating results

2

Understanding roles in ActivityInfo

Understanding Roles in ActivityInfo

Determining User Permissions and Actions

- Roles determine user permissions and actions within your database
- Create Roles to control data access and user actions

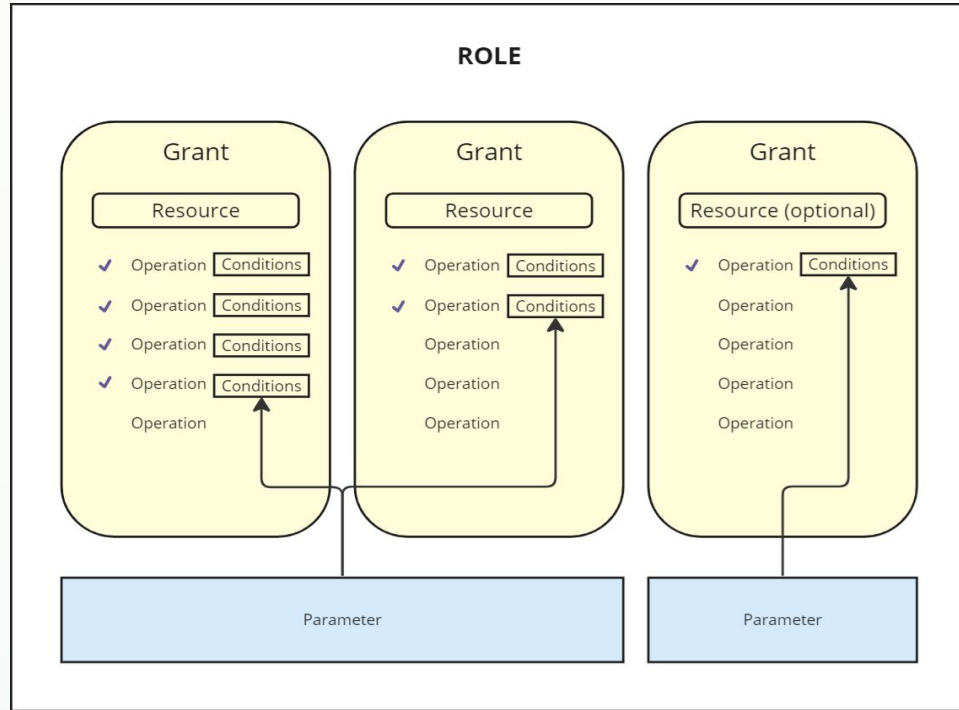
The image displays three panels from the ActivityInfo interface:

- Roles Panel:** A list of roles including Administrator, Case worker, Supervisor, and Case Observer. A green box highlights the Case Observer role, and a purple box highlights the 'side panel' label.
- Case Observer Role Detail Panel:** Shows options for 'Rename role', 'Duplicate role', and 'Delete role'. Under 'User management', there are checkboxes for 'Manage users' and 'Manage roles'. A green box highlights the 'Resources' and 'Parameters' tabs, with a 'Grant resources' button below.
- Permitted operations Panel:** Lists permissions for the Case Observer role, such as 'View all records', 'Display in the list of forms', 'Add any record', 'Edit all records', 'Edit 'Reviewer only' fields', 'Delete any record', and 'Bulk record delete'. A green box highlights this list.

Labels with arrows point to the 'Roles' list, the 'Operations and conditions' list, and the 'Grants and parameters' section.

Understanding Roles in ActivityInfo

Roles=Combinations of Grants and Parameters.



Understanding Roles in ActivityInfo

Key Concepts

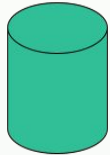
- Resources: Forms, Folders, Reports, and Databases.
- Operations: Actions performed on resources and users.
- Grants: Identify resource-specific operations.
 - Grants are inherited and can be overridden.
- Optional Grants: Enable flexibility in permission assignment.
- Conditions: Define rules for user operations.
- Parameters: Assign attributes to users for conditions.
- Roles: Combinations of Grants and Parameters.

Understanding Roles in ActivityInfo

Key Concepts

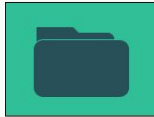
Resources

Databases



Database

Folders



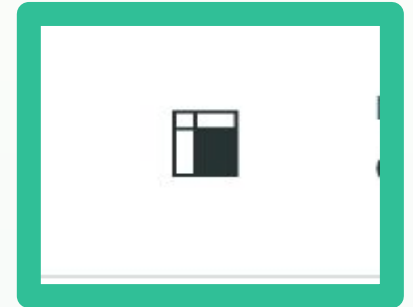
Folder

Forms



Form

Reports



Understanding Roles in ActivityInfo

Key Concepts

Operations = Actions performed on resources and users

Edit permissions

- View all records
- Display in the list of forms
- Add any record
- Edit all records
- Edit 'Reviewer only' fields
- Delete any record
- Bulk record delete
- Export records

Management

- Manage translations
- Manage record locks
- Manage collection links
- Audit user actions

Design

- Add forms, folders and reports
- Edit forms, folders and reports
- Delete forms, folders and reports

Sharing/ publishing

- Publish reports

Understanding Roles in ActivityInfo

Key Concepts

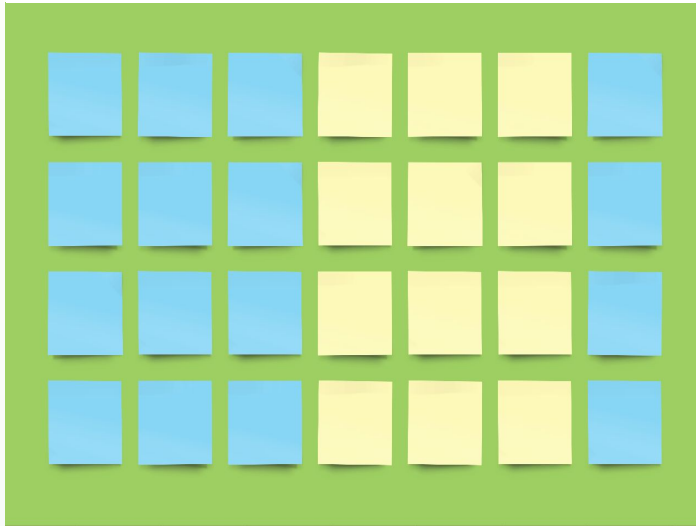
Grants: Identify resource-specific operations.

- A Grant identifies the specific resource on which the operations a user is allowed to perform are applied.
- Grants can be applied to any type of resource.
- Grants are inherited and can be overridden.
- Grants are inherited by all contained resources.
- A grant that is applied to a folder applies the settings to all the forms/folders within that folder, while a grant that is applied to an entire database applies the settings to all the resources within that database.
- Inherited grants can also be “overridden” on a contained resource by applying a new grant on that resource, if desired.

Understanding Roles in ActivityInfo

Cards Analogy

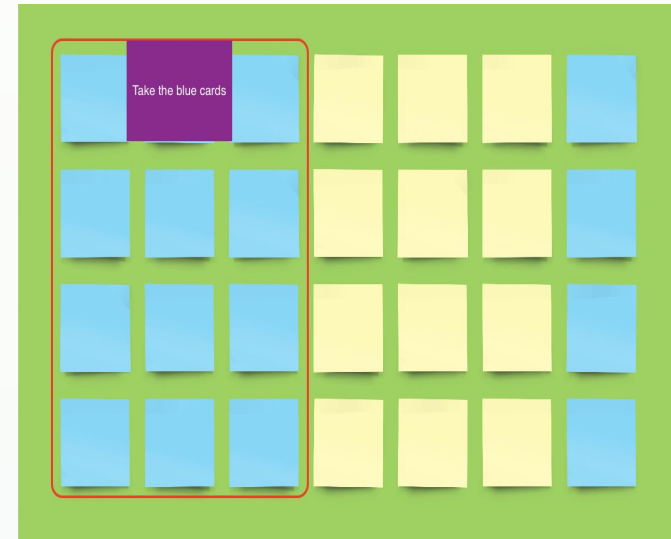
Resources



Operations

- Take
- Tear
- Colour
- Remove

Grants



Understanding Roles in ActivityInfo

Key Concepts

Grants, operations and reference forms.

To ensure that the visibility of reference forms is restricted, you must go to your database design, select the reference form and untick the box below in your list of operations



Permitted operations

BASIC

- View where conditions are met
- Display in the list of forms
- Add records where conditions are met
- Edit records where conditions are met
- Edit 'Reviewer only' fields
- Delete records where conditions are met
- Bulk record delete

Understanding Roles in ActivityInfo

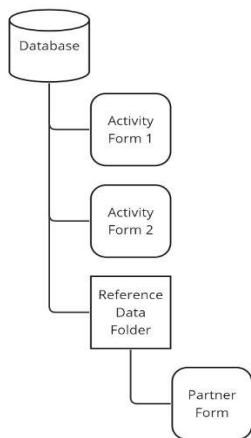
Key Concepts

Resources, Grants, and operations tied together:

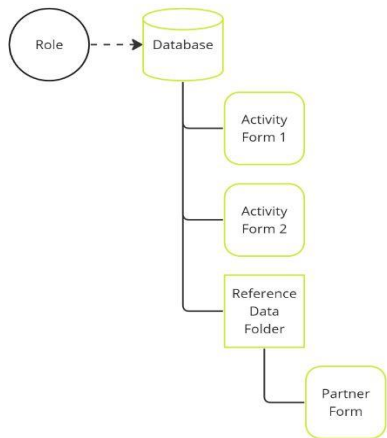
- Here is a form= identifying the resource
- Project supervisors can edit, view, and delete in forms=identifying operations (actions)that project supervisors can do
- Project supervisors can edit,view and delete records in the WASH cluster or in the entire database=Grants

Understanding Roles in ActivityInfo

Key Concepts-Inheriting grants

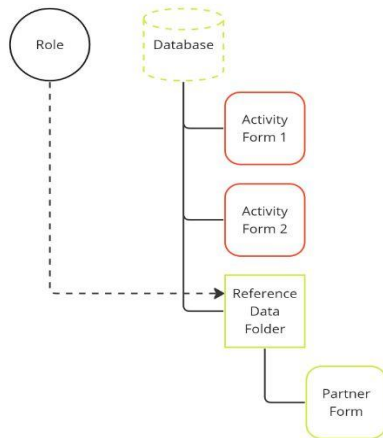


Say you have a database setup like so

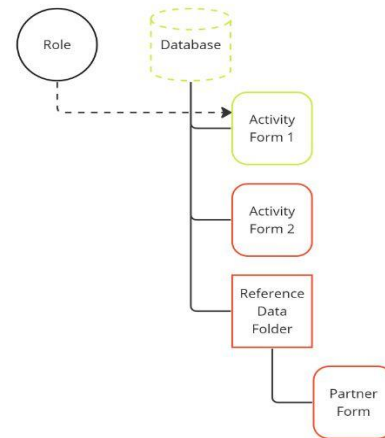


On my role, I apply an **explicit** grant to the database.

In doing so, the contained resources (folder, forms) also inherit this grant and the operations are applied to those resources as well. We can think of this as an **implicit** grant.



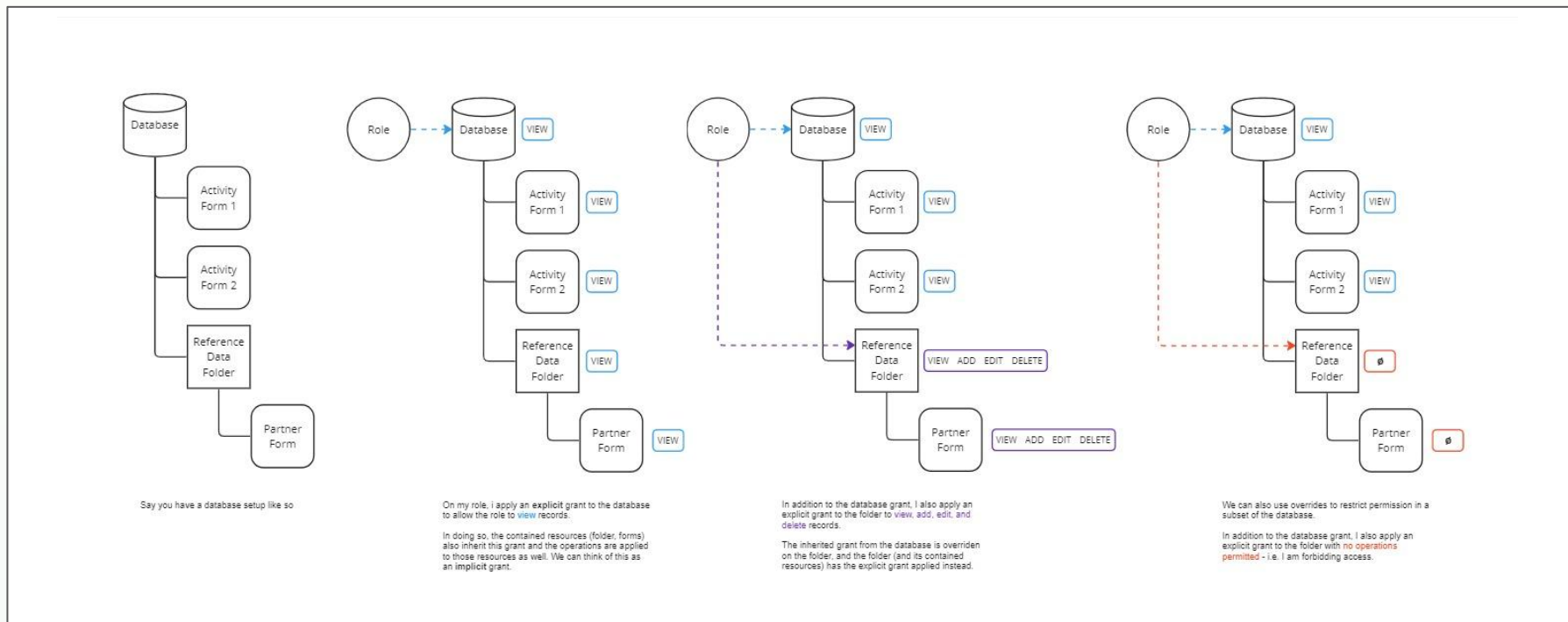
If I apply my explicit grant to the folder instead, then the contained resources inherit the grant in the same way. Note that I do not receive permissions on the root level forms as a result, as there is no grant on the database.



If I apply my explicit grant directly to a form instead, then the grant is only applied to the form. Note that the other resources in the database do not receive permissions as a result, as there are no other grants.

Understanding Roles in ActivityInfo

Key Concepts-Overriding grants



Understanding Roles in ActivityInfo

Key Concepts

- **Optional Grants:** Enable flexibility in permission assignment.
 - A grant can be set as optional, which means that you can choose whether to enable the grant for each user that you invite to your database.

Understanding Roles in ActivityInfo

Key Concepts: Conditions

- Conditions enable you to define rules that determine which records a user can perform operations on.
- Rules are always expressed as a formula that evaluates to TRUE or FALSE.
- The rule builder helps you write the most common kinds of formulas including:

Set Conditions

Allow when any of the following apply

[+ Add rule](#) [- Delete Condition](#)

Understanding Roles in ActivityInfo

Key Concepts

- The rule builder helps you write the most common kinds of formulas including:
 - determining whether a record is related to a **parameter**
 - determining whether a specific field matches a **specified value** or set of values
 - determining whether a record is assigned to the **user**

you can always use the formula editor to write your own formulas, including those that make reference to related fields or subrecords

The screenshot shows the rule builder interface. At the top, it says "Allow" followed by buttons for "View", "Add", "Edit", and "Delete". To the right of these buttons is a "when" dropdown menu currently set to "any", followed by the text "of the following apply". Below this are two rows of formula conditions. Each row consists of a dropdown menu with "Formula is true" selected, a text input field labeled "Enter formula", and a red "Delete" button. At the bottom of the interface, there are two buttons: "+ Add rule" and "Delete Condition".

Understanding Roles in ActivityInfo

Key Concepts-Parameters

- Parameters are attributes that are assigned to a user which can be used in conditions to control the record-level operations they are allowed to perform.
- Parameters are linked to a reference form, such as "Regions", which provides the possible values that can be assigned to a user.

Add parameter

Parameter ID
Provide an id for this parameter. This id can be used in formulas to reference the user's assigned parameter value (e.g. @user.partner).

Parameter Label
Provide a user-readable label for this parameter.

Possible Values
Select a form which provides the possible values for this parameter when a user is assigned to this role (e.g. Partner form).

DATABASE 2023 Wash Activities	▶	FORM Distribution of Soap and detergents
DATABASE 2023-08-15 Permission Demo (MPR)	▶	FORM Users

3

How to create roles in ActivityInfo

How to create roles in ActivityInfo

Steps involved in adding a Role

Before you start

- Make sure you have already added a database and
- You have been assigned to a role with the “Manage roles” and “Manage users” operation permitted.
- If not, for practice purposes, you can use the training and monitoring template to add a new database.

1. Add the role
2. Add resources to the role
3. Grant a resource
4. Assign operations
5. Save and invite a user

How to create roles in ActivityInfo

Example 1: Limiting access to records based on a Parameter

- Scenario: Program staff serving beneficiaries by region.
 - Ensure staff can only access records in their assigned region.
- Create a Role for "Programme Officer" with parameters and conditions.
 - Create a single role
 - create a parameter
 - Assign Grants
- Benefits of parameters in this scenario

How to create roles in ActivityInfo

Example: Limiting access to records based on assigned user

- Scenario: Case management
 - Case records contain highly sensitive personal information
 - Each case assigned to a single case worker
 - Each case worker has a supervisor for oversight
- Role requirements
 - Case workers to access only records of the cases they are assigned to
 - Supervisors can access their own cases as well as those that belong to their case workers
 - Case workers should be able to view reference records in data entry but not be able to edit them
- Role configuration
 - Add **condition** based on user assignment
 - Add **parameter** for Supervisor assignment
 - **Grant** to Reference folder with *View records* permission only

4

Best practices for designing roles

Best practices for designing roles

Design roles with **narrow** user permissions



Employ the Principle of Least Privilege as a rule



To maintain integrity: consider whether users really need the ability to **Edit** or **Delete**



To protect sensitive data: be careful with **Export** and **Publish** permissions



Grant **User Management** operations selectively



Edit permissions

Permitted operations

BASIC

- View all records
- Add any record
- Edit all records
- Edit 'Reviewer only' fields
- Delete any record
- Bulk record delete
- Export records

DESIGN

- Add forms, folders and reports
- Edit forms, folders and reports
- Delete forms, folders and reports

MANAGEMENT

- Manage reference data
- Manage translations
- Manage record locks
- Manage users
- Manage roles
- Manage collection links
- Audit user actions




SHARING AND PUBLISHING

- Share reports
- Publish reports

Cancel Save

Best practices for designing roles

Design roles with the **broad** context in mind

-  **Understand** roles and responsibilities within your organization
-  **Educate** end users on data security
-  Regularly **review** and **update** roles when the context changes

Best practices for designing roles

How many roles do I need?

You will likely need to create fewer roles than you think.

- Design roles according to the **tasks** that a group of users needs to do, then
 - use **optional grants** to enable access to different resources while maintaining a common set of universal permissions
 - use **parameters** to differentiate which records a user can work with

| 5
Q&A

Up next

OCT 19

SESSION 3

Office Hours -
Designing Roles in
ActivityInfo

What we'll do:

- Discuss questions and best practices for designing roles in the ActivityInfo based on real examples and your own use cases.