

# Six steps to creating a scalable, robust and flexible BOXI Security Model"

Allanna Firth - Technical Services Director



# Who am I & Why BOXI security?

## Who am I?

- BiTS is owned by Marcel Abraham and myself
- Marcel Abraham heads up Training Services, started 2004
- I head up Technical Services, started 2006
- 9+ Years working with Business Objects version 3 > 4> 5>6> XI
- 6+ Years consulting for blue chip companies
- BI Team Lead for first BOXI R1 Unix implementation in Australia 2.5 years ago
- Since then worked on numerous BOXI client sites doing migrations, installations, configurations, environment reviews, training, universe design, application development as well as data design & etl.

## Why BOXI Security Model presentation?

- So many people on client sites ask questions relating to security, inheritance rules and application access
- Have seen many migrations where the security model has not been reviewed
- Problems occur after fitting and forcing old security model into new
- Good chance to help understanding
- Get people thinking - about clearing out old practices, refreshing/updating requirements and setting up new security model
- Get the most out of BOXI, have a security model that is not a maze and hard to maintain, or that one person understands make it organised and structured
- BOXI Security Model is a piece of work or mini project in it's own right
- Should be part of all BOXI pre-installation and/or pre-migration planning efforts



# To start...

The **six steps** I'll be taking you through use the development lifecycle BiTS use for any piece of work

In terms of BOXI Security Models and my experience;

**Step 1 ~ Analyse**



**Step 2 ~ Design**



**More focus is needed here**

**Step 3 ~ Build**

**Step 4 ~ Test**

**Step 5 ~ Implement**

**Step 6 ~ Support**



# Step 1 ~ Analyse

## Step 1 ~ Analyse

- Understand what your security model needs to do to meet your IT /Business/User requirements
- Understand what BOXI can do from a security and BI tool functionality perspective
- Understand the BOXI security components and how they interact

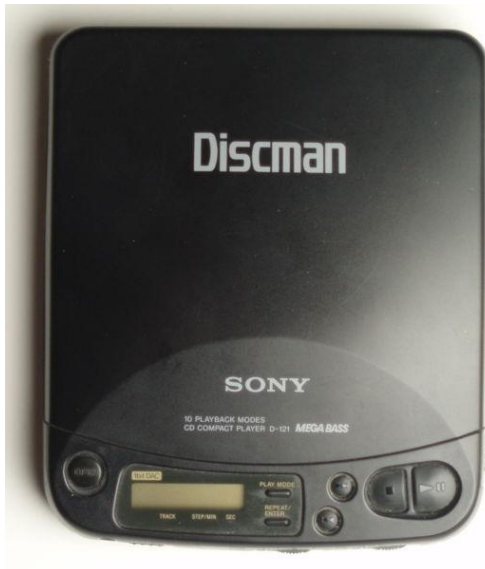


# Step 1 ~ Analyse

What does your security model need to do?

Before you can answer this question...

**You need to understand what Business Objects XI can do?**  
And if using older version - how it is different ...



# Step 1 ~ Analyse

## What can BOXI do?

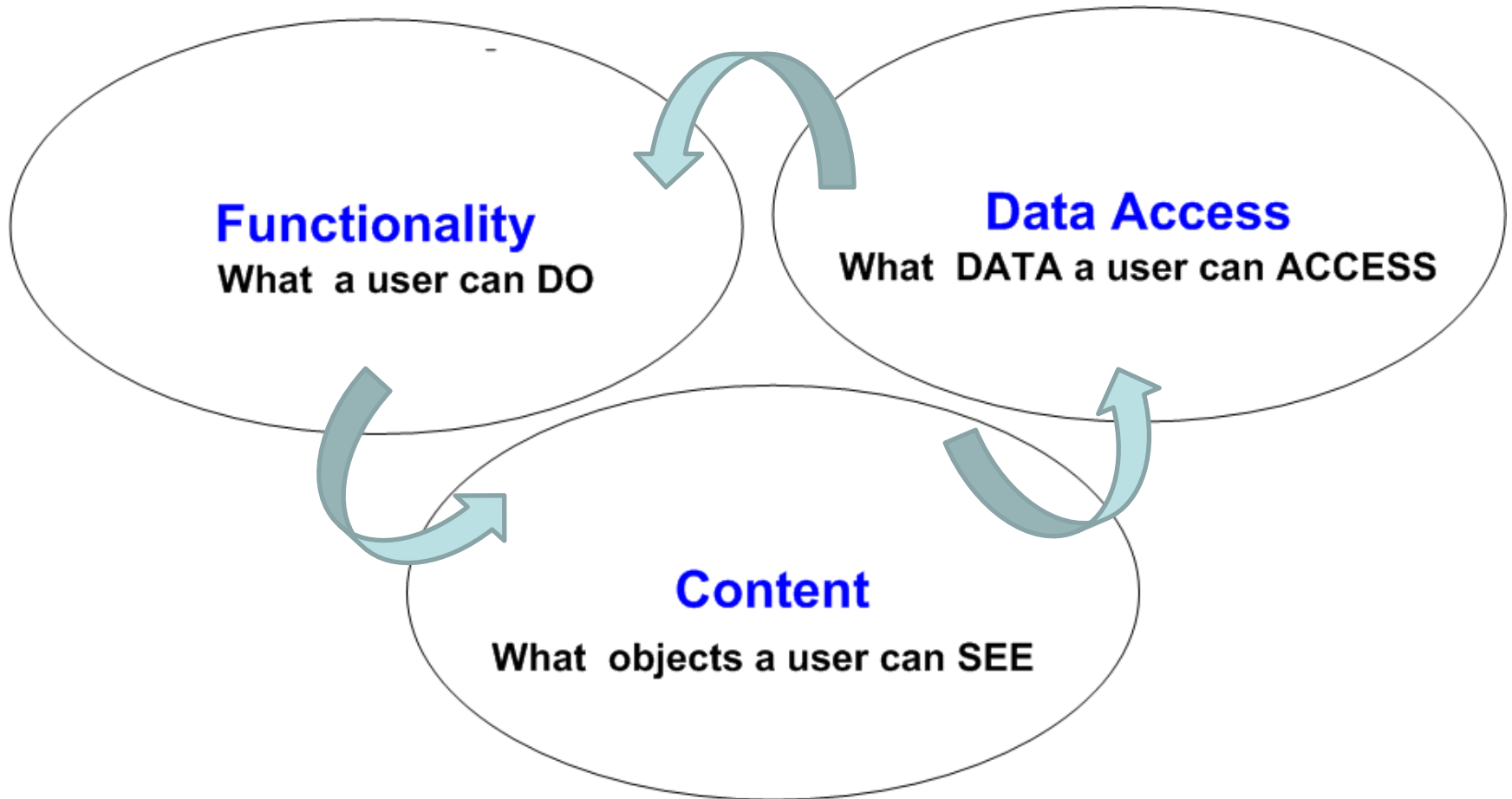
(How you can implement security if you don't know the full capability or functionality of the tool)

- What functionality does BOXI have for creating reports? performing analysis? Publishing?
- How does the tool work? What is the back end or architecture like?
- What security can be applied in BOXI ?
- What are the key differences for security with the newer version of Business Objects?



# Step 1 ~ Analyse

What does your BOXI Security model need to do?



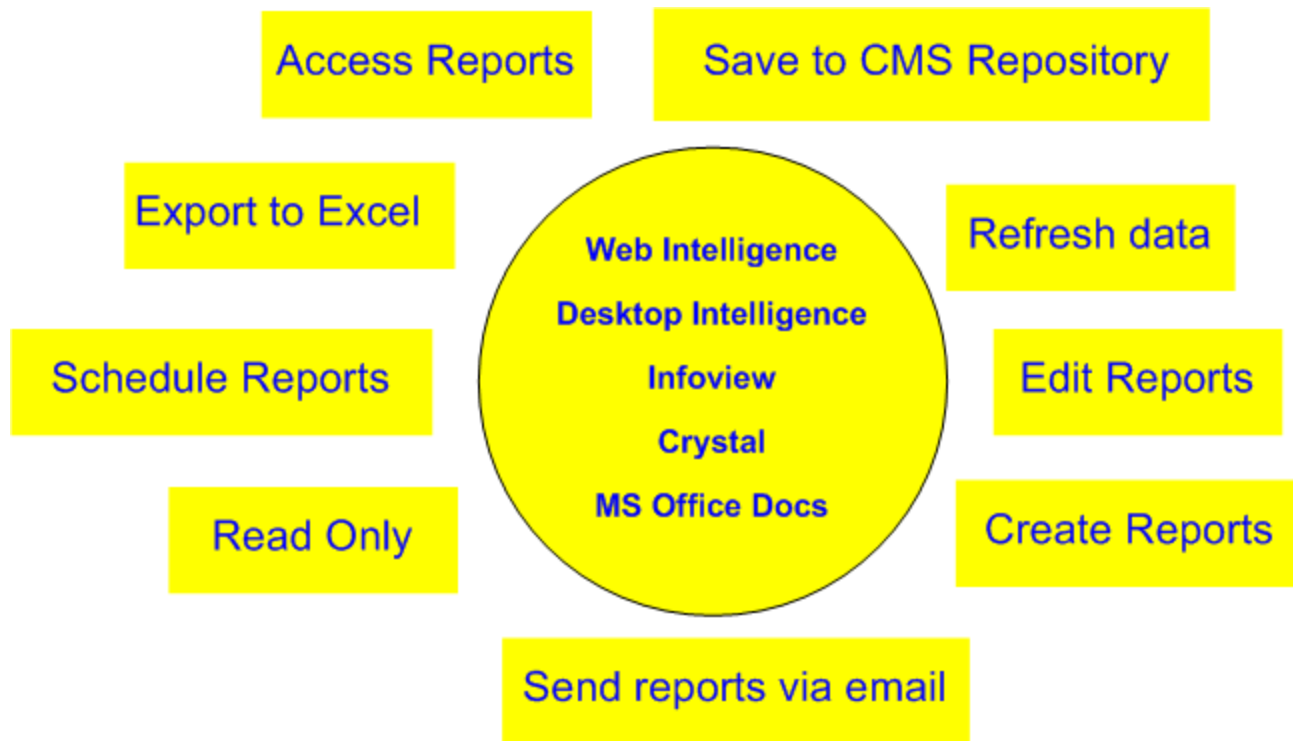
# Step 1 ~ Analyse

What does your security model need to do? **Functionality**

What do the users want to do with **BOXI** ?

What do they do currently? What do they want to do? What does IT want them to do?

What does the business want to do? Where is the middle ground?

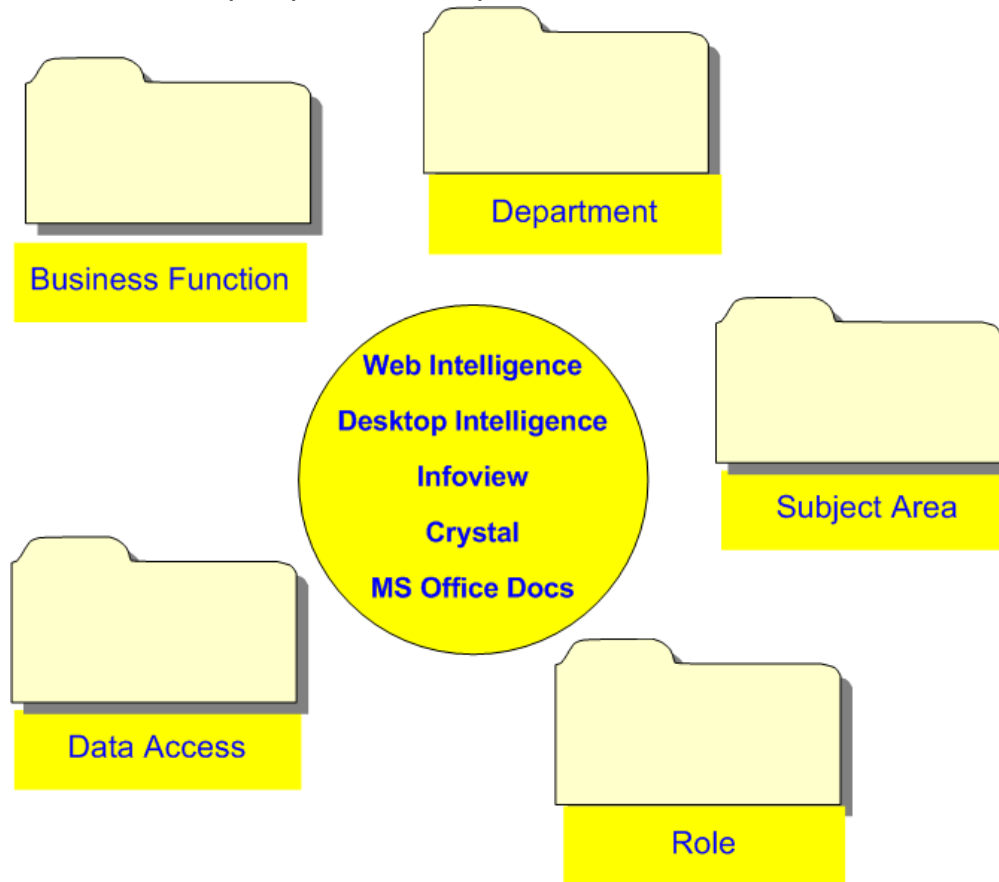


# Step 1 ~ Analyse

What does your security model need to do? **Content**

What do the users need to see? **Objects** – folders, report objects, universes & data

Who decides? Business and/or IT? With multi dimensional analysis harder to keep information in isolated department / business function buckets - most kpi reports made up of a mixture of data - new folder structures



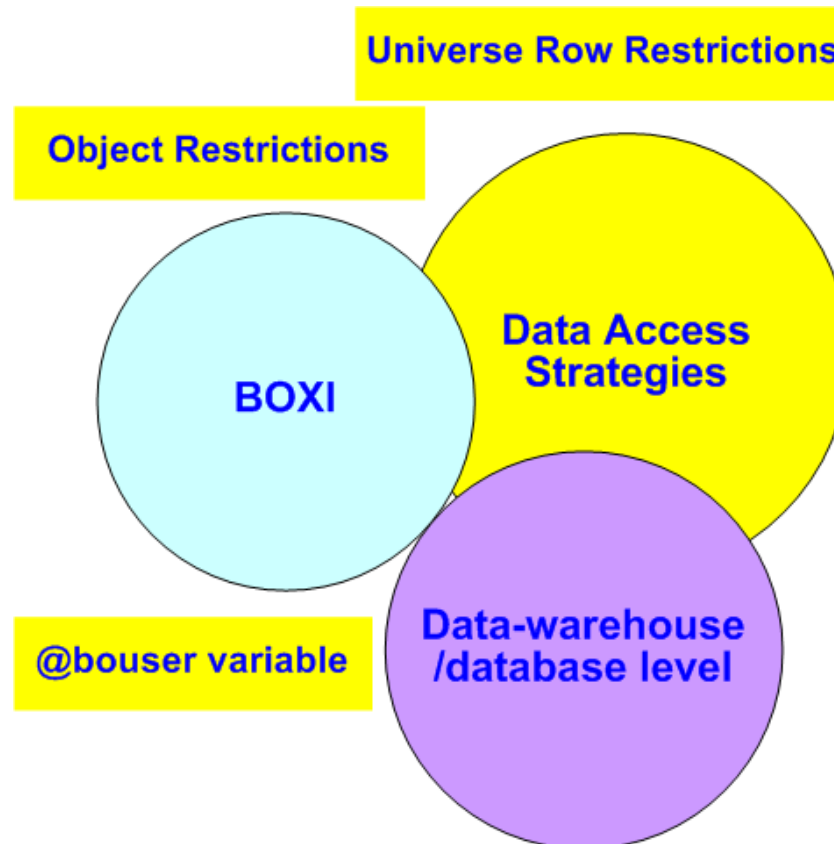
# Step 1 ~ Analyse

What does your security model need to do? **Data Access**

What data does the user need to access?

Data access can add another layer of complexity to hierarchies of groups and folders if not thought about

Only restrict what you need to, consider other data access strategies e.g @bouser variable & database level

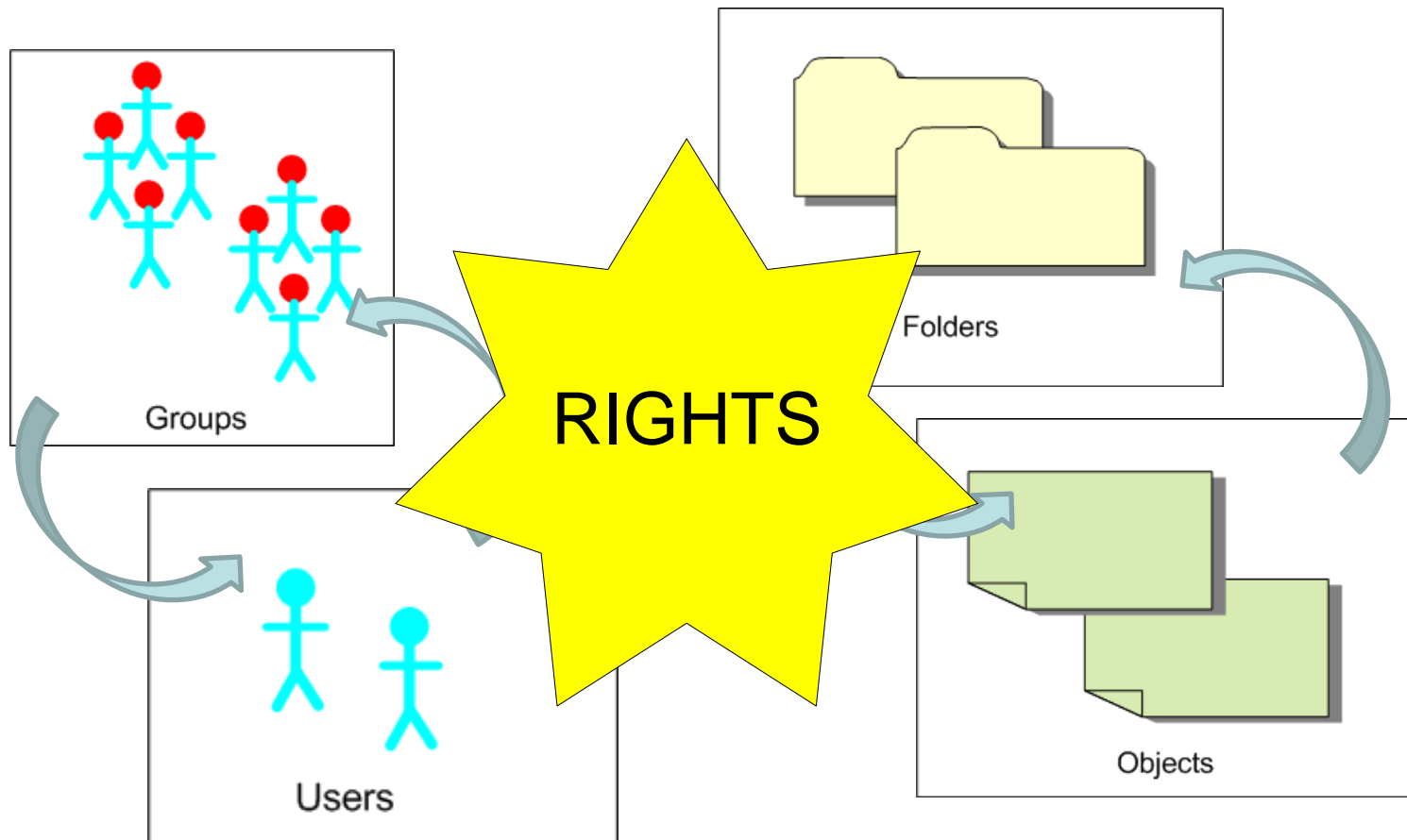


# Step 1 ~ Analyse

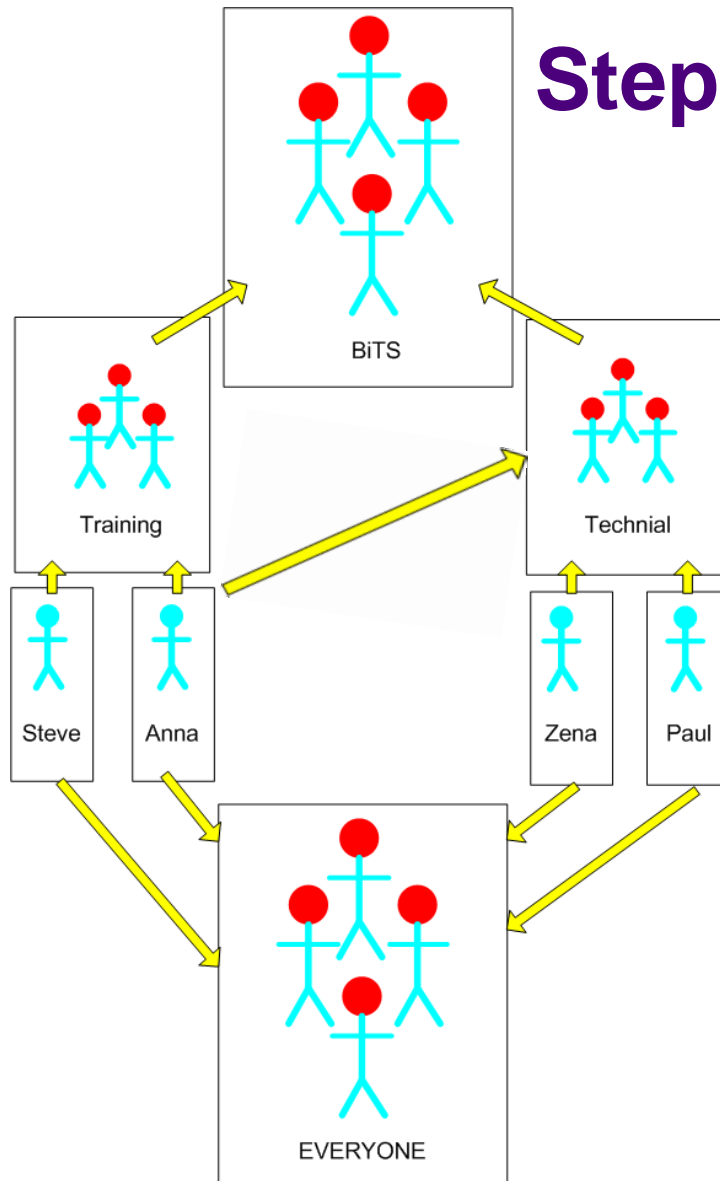
What are the BOXI Security Model components?

BOXI security model components are; **Groups, Users, Folders, Objects & Rights**

**Rights** are the lynch pin between the components



# Step 1 ~ Analyse



## Groups & Users

- **All users** belong to the default group **Everyone**
- **Users** can belong to **multiple groups**
- **Groups and Users** have **rights assigned to objects** (folders are objects as are documents)

**Groups :** BiTS, Training, Technical, Everyone

**User :** Zena

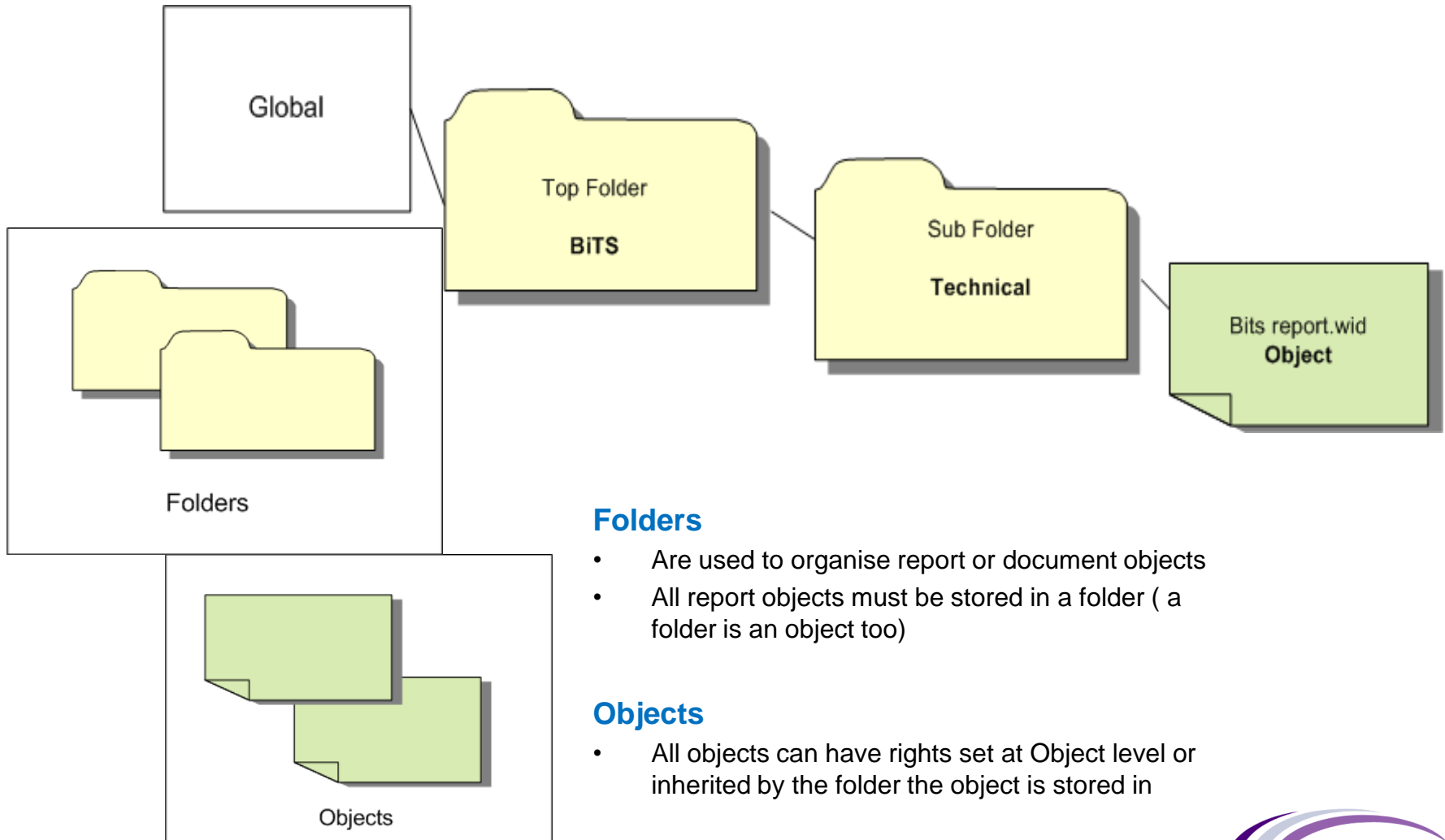
**Groups:** BiTS , Technical, Everyone

**User:** Anna

**Groups:** BiTS, Technical, Training, Everyone



# Step 1 ~ Analyse



## Folders

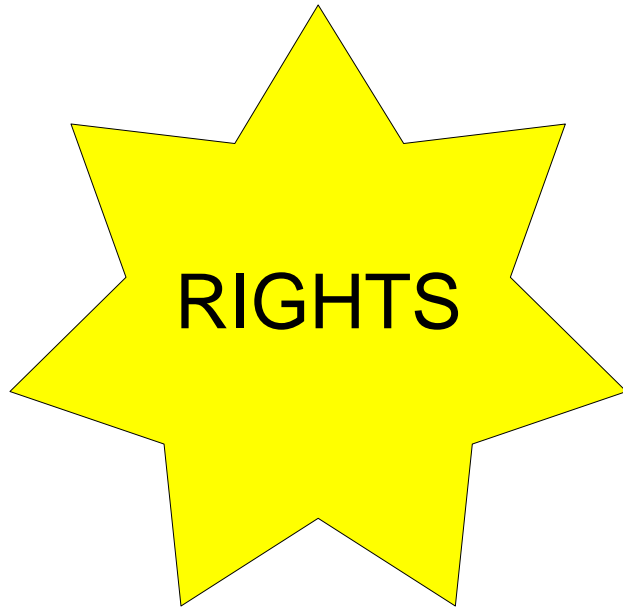
- Are used to organise report or document objects
- All report objects must be stored in a folder ( a folder is an object too)

## Objects

- All objects can have rights set at Object level or inherited by the folder the object is stored in



# Step 1 ~ Analyse



- are the **base unit** for controlling **USER ACCESS**
- specify individual actions a user can perform on an object
- can be set on user access for ;
  - folders**
  - report objects
  - document objects**
  - program objects
  - other BOE objects**

To **secure content** published to BOXI **rights** can be set for each object

Security flows down from **Global > Folder > Object**

**Global**

**Folder**

**Object**

Two types of User Access methods

- 1) **Predefined**
- 2) **Advanced**



# Step 1 ~ Analyse

## Predefined

### No Access

### View

### Schedule

### View on Demand

### Full Control

## Advanced

### Predefined

Not all access levels are available for all object types  
Access levels available are determined by the object's function  
Schedule rights exist for a report object, but it doesn't make sense for a user object

### No Access

The user or group is not able to access the **object** or **folder**

### View

Set at the folder level, the user or group is able to view the folder, the objects contained within the folder, and all generated instances of each object.

### Schedule

Schedule to different formats & destinations, set parameters & database logon information, pick servers to process jobs, add contents to folder, copy object or folder

### View on Demand

In addition to rights granted by at Schedule access level, user gains right to refresh data\ "on demand" from data source

### Full Control

Grants all available advanced rights & allows users to view object & modify properties  
Allows users to delete objects (folders, objects, and instances).  
Allows users to modify all object's properties, including object rights set on folder or object.

### Advanced access levels

Complete control over object security / granular object rights  
Allows customisation by setting specific rights the user or group can perform



# Step 1 ~ Analyse

Advanced

Explicitly Granted

**Explicitly Granted**

User or group is given designated access right

Explicitly Denied

**Explicitly Denied**

User or group is not given designated access right

If user or group is granted the access right through another group membership denial takes precedence

Not Specified

**Not Specified**

Right is not assigned to user or group at any level, so it is not granted. Unlike an explicitly denied access right, user or group could be granted access right through another group membership, or inherit the rights from a higher group or folder level

N.B Denied Right Overrides Granted Right

## Calculating Effective Rights :

**Explicitly Denied (D)** wins over **Explicitly Granted (G)** ( $D + G = D$ )

**Explicitly Granted (G)** wins over **Not Specified (NS)** ( $G + NS = G$ )

**Explicitly Denied (D)** wins over **Not Specified (NS)** ( $D + NS = D$ )

**Explicitly Denied (D)** wins over **Granted (G)** and **Not Specified (NS)** ( $D + G + NS = D$ )



# Step 2 ~ Design

## Step 2 ~ Design

- Design a security model that meets IT/Business/User requirements based around simplicity and practicality
- Design your security model based on Business Objects best practice recommendations
- Design your security model structure to work within BOXI's new technology and security framework - will you use single sign on , third party authentication?
- Take the time to plan & think about design - will it meet different scenarios /changing requirements?



# Step 2 ~ Design

## Design your security model based on Business Objects best practice recommendations

Good practice to organize security around functionality and content

“XI R2 this model works even better, due to the availability of two parents — not available in version 6x, whose strict tree structure means that a user can be in multiple groups, but a group can't have two Parents”.

“Business Objects recommends that you structure your system's security around two axes:

- What a user can do
- What content a user can access

*Extracted from 'BusinessObjects 6.x to XI Release 2 Migration Guide'*



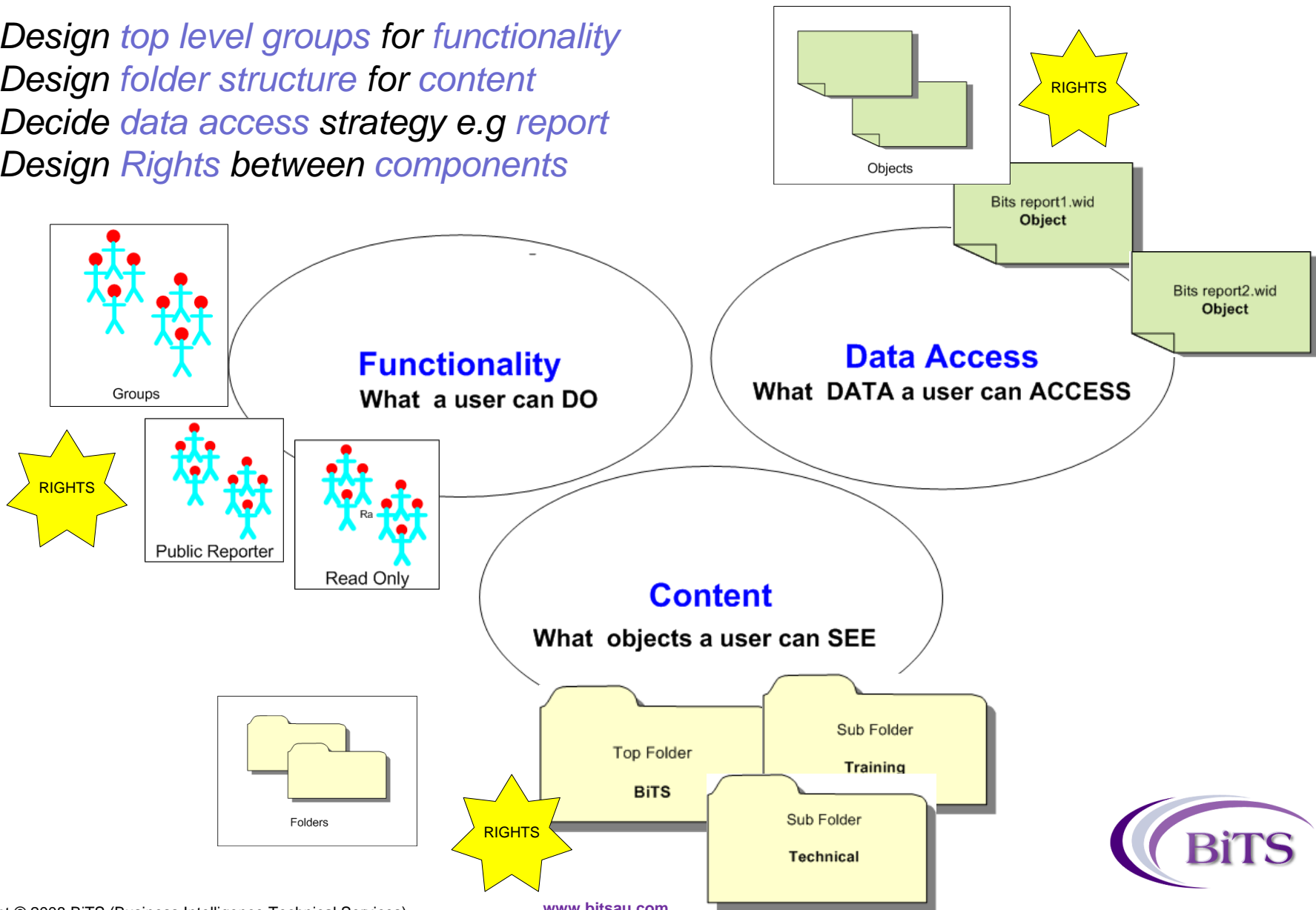
# Step 2 ~ Design

Design top level groups for functionality

Design folder structure for content

Decide data access strategy e.g report

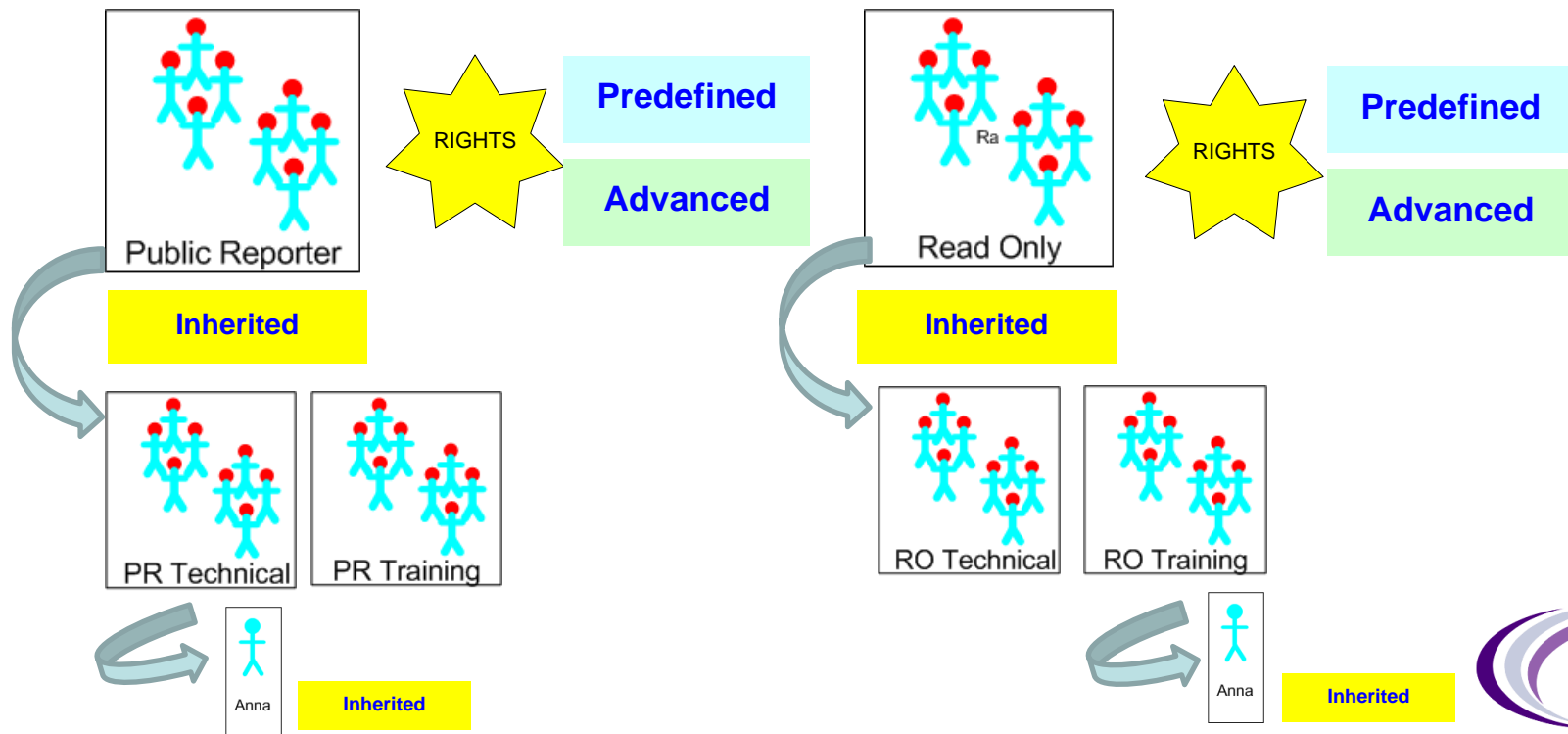
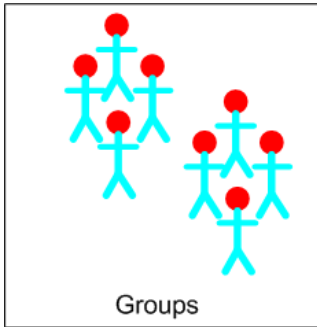
Design Rights between components



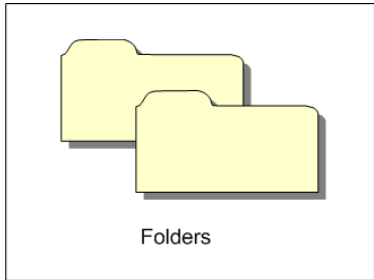
# Step 2 ~ Design

## Groups:

*Review & choose rights at parent (top) group level*  
*Mirror folders as sub groups*  
*Inherit rights to sub groups*  
*Add users to sub groups*

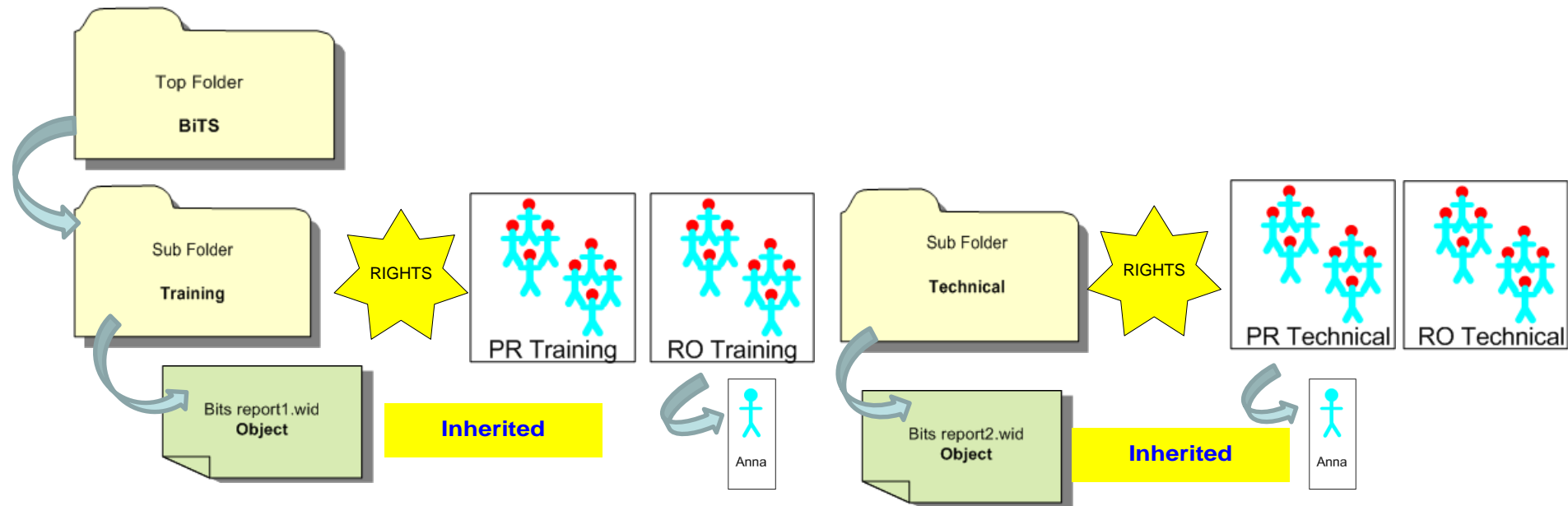


# Step 2 ~ Design



## *Folders:*

*Give access to relevant sub group on appropriate folders  
Choose inherited - predefined - advanced rights at folder level*



# Step 2 ~ Design



## Rights Design Recommendations:

- **Assign security at the folder level to groups whenever possible**
- **Avoid setting rights for specific users on specific report objects**
- **Reduce the complexity of your security model – keep it simple not too many levels**
- **Use Predefined Access Levels whenever possible because it reduces the complexity of the security model.**
- **Grant the Everyone group No Access at global level to restrict access to the system.**
- **Grant specific rights to appropriate groups**
- **When securing reports and documents based on Universes or Business Views, ensure users have appropriate rights to Universes or Business Views**



# Step 2 ~ Design

Based on previous scenarios

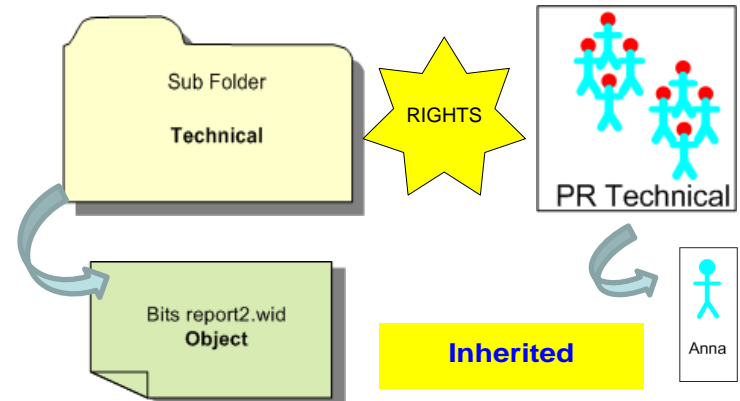
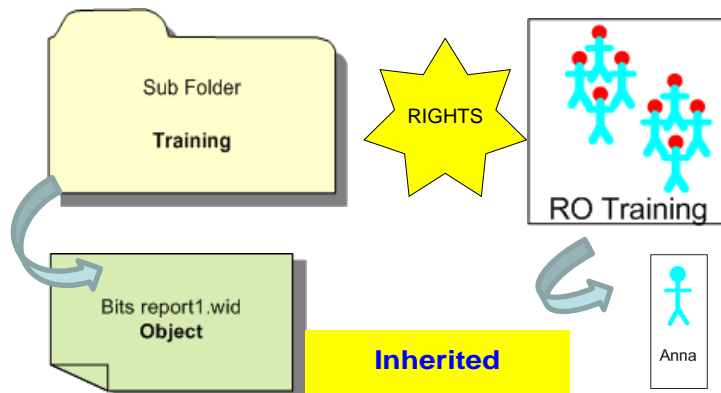
User **Anna**:

*Has “read only” rights on the folder **Training***

*Inherited “read only” rights on the report object **Bits report1.wid***

*Has “public reporter” rights on the folder **Technical***

*Inherited “public reporter” rights on the report object **Bitsreport2.wid***



# Step 3 ~ Build

## Step 3 ~ Build

- Build a skeleton or small part of your security model in test environment
- Choose the group, users, folders, objects and rights that are a good reflection sample of overall needs to create
- Set up groups, users, folders and objects, modify rights at group and folder level via CMC



# Step 4 ~ Test

## Step 4 ~ Test

- Test the security model is meeting IT/Business/User specified requirements
- Make sure users and groups
  - (i) only have functionality they should
  - (ii) can see folders /objects they are allowed to and
  - (iii) access data they are supposed to
- Test scenarios for changing Business /IT/User needs
- Is the model flexible? Scalable? If not, this is your chance to tweak and revise by going back to steps. 1 and 2.
- 



# Step 5 ~ Implement

## Step 5 ~ Implement

- Implement security model into BOXI production environment
- Create/ import setup in CMC groups, users, folders, objects
- Set access rights via CMC on groups, folders and application level

# Step 6 ~ Support

## Step 6 ~ Support

- Train end users to understand standard profiles, functionality and content management
- Train IT /business/user to administer users, groups functionality and content management
- Review security model periodically in line with changing IT/Business/User Requirements



# To finish..

## Questions & Answers

For more information about this presentation or a copy please email [Allanna.Firth@bitsau.com](mailto:Allanna.Firth@bitsau.com)

Thankyou

