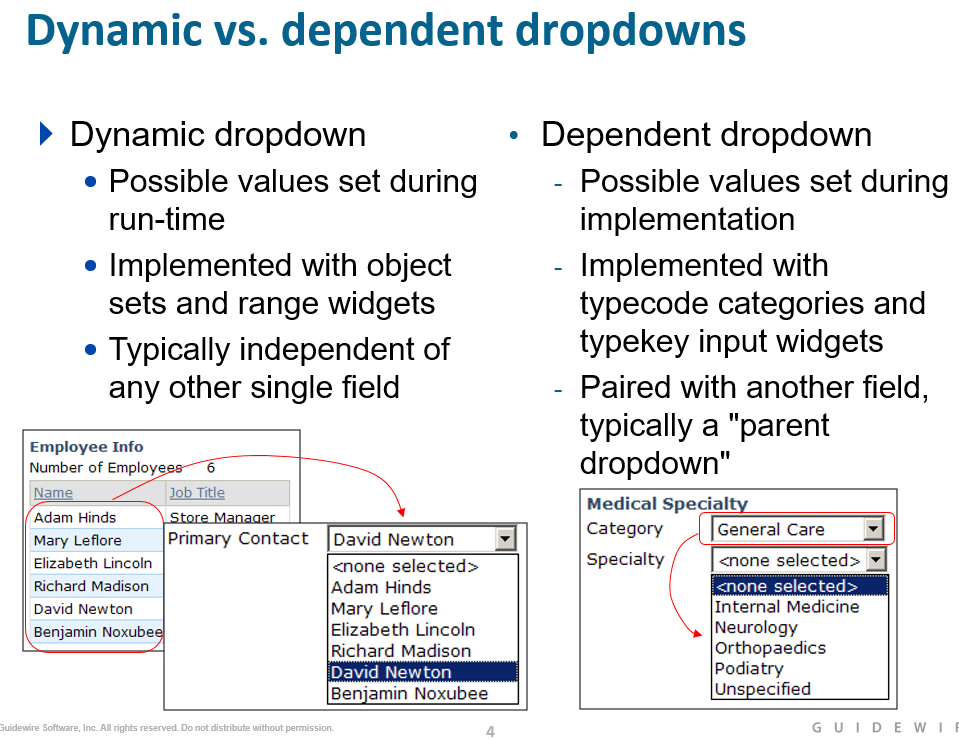
**Interview Questions by Different Companies**

**L&T Interview Questions**

1. **Difference between static type list and dynamic type list?**



* Dynamic dropdowns created using:
  + Range input widget (for detail views)
  + Range cell widget (for list views)
* Range widgets include these properties:
  + valueRange - to specify list of values
  + value - to store selected value

**Q) When would you implement a dynamic dropdown using a range input (as opposed to a range cell)?**

You would use a range input when adding a dynamic dropdown to a detail view; range cells are used for adding dynamic dropdowns to list views

1. **How can you do the LOB Mapping?**

You model your business structure by configuring the relationships between a set of typelists:

• LossType and LOBCode – Model which business units cover each kind of loss.

• LOBCode and PolicyType – Model the kinds of policies written by each business unit.

• PolicyType and CoverageType – Model the types of coverages that policies can contain.

• CoverageType and CoverageSubtype and ExposureType – Model the kinds of exposures that are compatible

with coverages.

**Workflows?**

Workflow elements Start, Finish, Enter, Exit, GO, TRIGGER, and TIMEOUT can all contain embedded Gosu. The

Workflow engine executes this Gosu code any time that it executes that element. The specific order of execution is:

The Workflow engine runs Start before everything else

• The Workflow engine runs Enter on entering a step.

• The Workflow engine runs Exit upon leaving a step. It runs Exit before the branch leading to the next step.

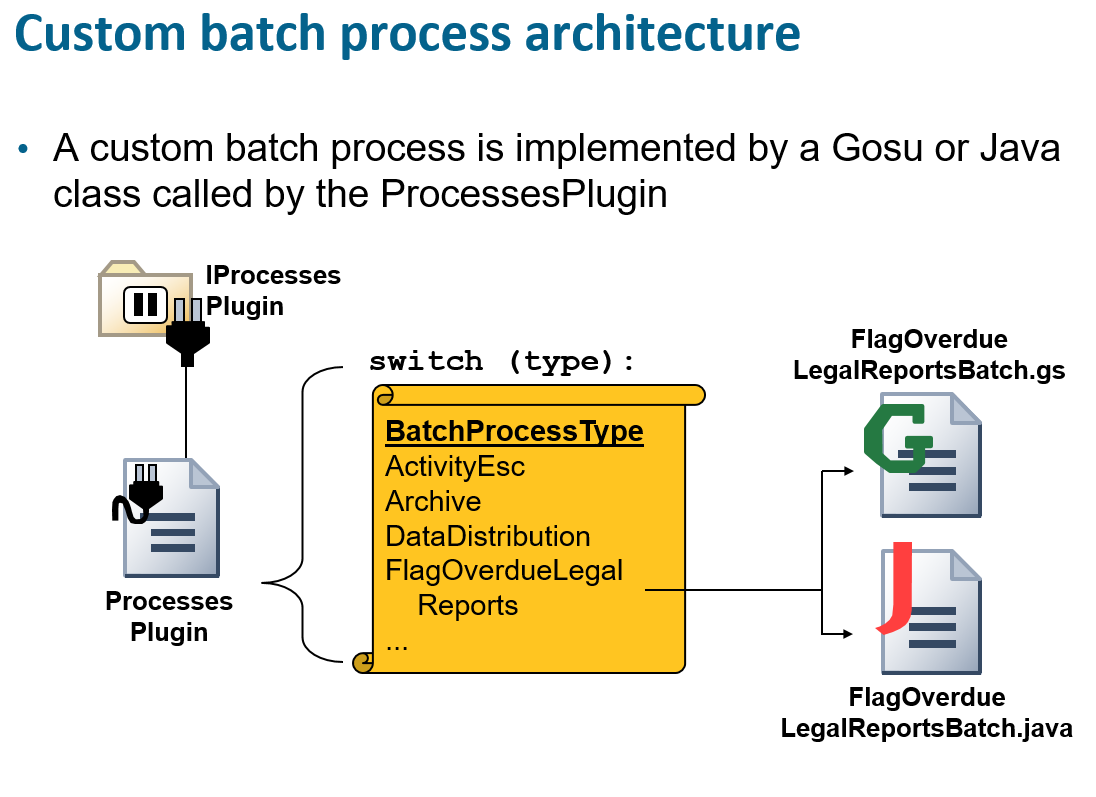
Thus, the actual execution logic from Step A to Step B is to Exit A, then do the Branch, then Enter B.

• The Workflow engine runs GO, TRIGGER, TIMEOUT elements as it encounters them upon following a branch.

• The Workflow engine runs Finish after it runs everything else

The Workflow editor is found only in BillingCenter, ClaimCenter, and PolicyCenter. The Workflow editor is used to modify workflows, which are complex and potentially asynchronous processes that execute business processes

1. **Custom batch process?**



* Batch process schedules are specified in **scheduler-config.xml.**

**Syntax**: <ProcessSchedule process="BTSRegulatoryReport">

<CronSchedule hours="0" minutes="05"/>

Steps to Implement Custom Batch Process:

* Add batch process to BatchProcessType typelist
  + Specify how batch process can be run
* Create new class that extends BatchProcessBase
* Implement or modify the ProcessesPlugin class to call the new batch process class
* Deploy the changes

BatchProcess::

BatchProcessType.ttx and mention the custom batch name

Your custom batch process must extend the BatchProcessBase Gosu class. You must override the abstract doWork method, which takes no arguments. You can override other methods, but it is not strictly necessary as the base class defines meaningful defaults.

The doWork method does not have a bundle to track the database transaction. To modify data, you must use the Transaction.runWithNewBundle(...) API to create a bundle.

Exclusive(){ return false}--it will allow multiple batches to process at a time.

IMaintananceToolsAPI Webservice widely to run the custom batches and properties loading.

1. **Have you worked on Plugin Delegator?**
2. **Have you consumed web Services? Have you published web services?**

**Publishing Web service Steps**

* Create the Gosu package and class
* Add the web service annotations : **@WsiWebService** annotation specifies the class is a WS-I web service
* **@WsiPermissions({"soapadmin"})**
* **@WsiGenInToolkit**

Specifies permission the user must have to use the web service (or any of its methods).

Defaults to "soapadmin" permission

* **@WsiAvailability(MAINTENANCE)**

Specifies the run level at which web service is first usable

* Add methods
  + Any method that modifies Guidewire data must create and commit a bundle
* Deploy your changes

**ValueMometum Interview Questions?**

**Difference between Input set and Detail view panel?**

**Detail views** typically have toolbars when at least some of the data in the detail view must be editable (and therefore Edit, Update, and Cancel buttons are needed).

Technically, a detail view is not required to have any input columns. A detail view with atomic widgets does need at least one input column, however, and virtually every detail view has atomic widgets.

Guidewire applications automatically put a vertical blue line in between input columns. This line can be seen in the screenshot above in between the two red rectangles

An **input set** is a set of widgets that can be reused in multiple detail views. Input sets give developers the ease of creating the widget set once and then referencing it wherever it is needed.

An **input set** is a named group of widgets can contain only widgets that can be put in detail views

* Like detail views, input sets can contain:
  + Inputs
  + Input dividers
  + Labels
  + Embedded list views
* Unlike detail views, input sets cannot:
  + Contain input columns

**Can we create check or issue payment without exposure?**

Answer: Yes, only Expense check not the medical one.

**What is matter? What is** Subrogation**?**

Ans: Matter is an entity. Subrogation status is a type key, which the holds the values as review, open, closed.

**What are the limits for reserves?**

Answer: If there is no Medical or Indemnity exposure, you do not see them on the Reserve worksheet. You have to specifically add the limits on each claim. Without reserve worksheet, you can only issue expense check.

**Can we refresh the policy every time?**

We cannot refresh the policy in these situations.

Refresh the policy will not work if the risk units/coverage does not exist on the NEW / Updated policy.  You cannot refresh if the risk unit/coverage does not exist on the new policy and there are financial transaction associated to the risk unit / coverage.

So, on P&C if a vehicle is removed from the policy - but we have a claim with that vehicle on the exposure and there are financial transactions on that exposure.  We cannot refresh

**What are the Field level validation we have in Guidewire?**

* **Field-level validation** is a validation behavior tied to one or more specific fields, which can be implemented at:
  + Data model level
    - Datatypes: Guidewire automatically prevents data of the wrong datatype from being saved and warns the user of the error
    - Field validators: Field validators declared in **fieldvalidators.xml**
  + UI level
    - Validation expressions:
    - Syntax: ***condition* ? NULL: *error Message*, or  
       *condition*? *error Message*: NULL**

**What is an exposure? How it’s created on Claim?**

**BLACKCOMB CONSULTANTS INTERVIEW Questions**

1. **FNOL Steps?**
2. **Claim Segmentation steps?**
3. **Explain Assignment rules logic?**

Default Group Assignment Rules, Global Assignment Rules

**Explain Subrogation**

1. **Explain Message transport Plugin, Reply Plugin, Response Plugin work?**
2. **What is Card View, List view, Detail View?**

* A **list view** is a container widget that typically displays a set of rows that are related to one object or one query
  + Users can view and, in some cases, edit the data
* **List view Panel : Which displays multiple records at the same time.**
* **Secondary Container : they organizes primary containers.**
  + **List Detail Panel : contains List on the Top and Detailed information on bottom.**
  + **Card Detail Panel : Is collection of cards. Which has one or more list view/Details view.**

1. **How Batch Process works?**
2. **How do you configure Batch process as UI Runnable?**

Under the plugin name In categories you need to mention **batchprocesstypeusage.ttx** as UIRunnable,API Runnable, Schedulable.

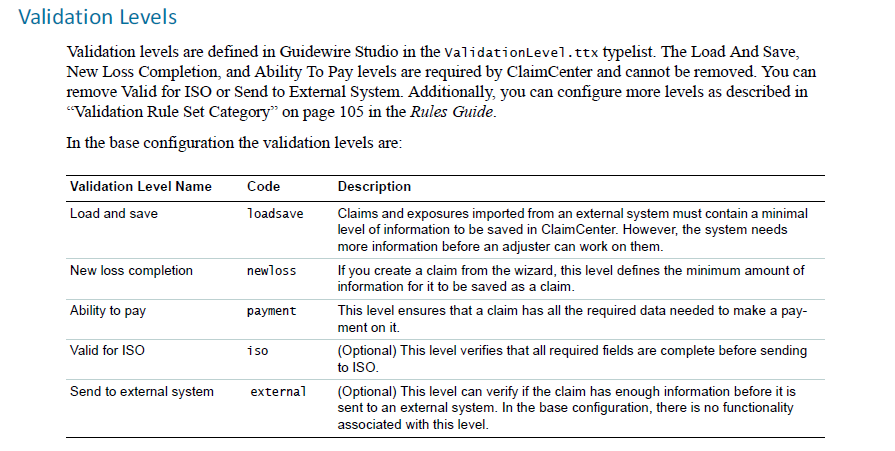
1. **Explain The cron Job syntax in Scheduler-config.xml?**

* Batch process schedules are specified in **scheduler-config.xml.**
  + **Syntax**: <ProcessSchedule process="BTSRegulatoryReport">
* <CronSchedule hours="0" minutes="05

1. **What methods we override in Message Transport IMPL?**

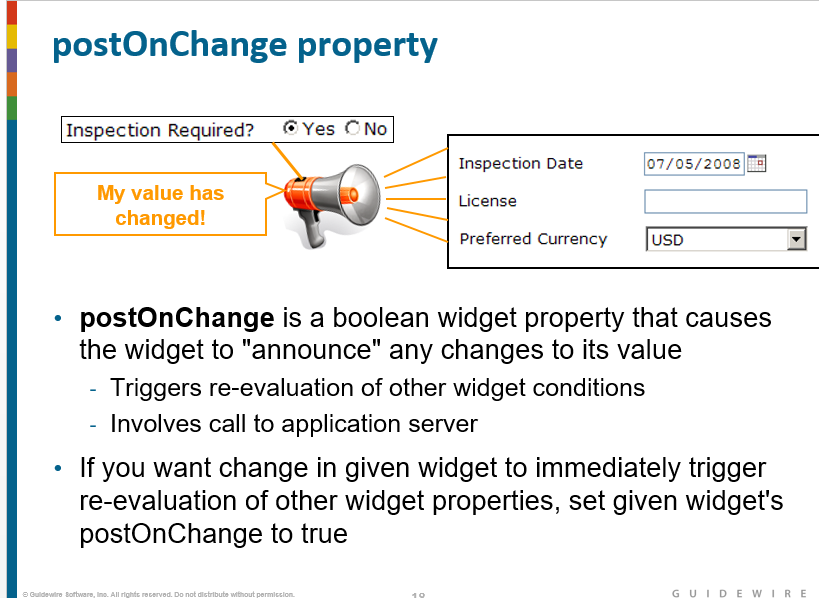
Eg: Public override function Suspend(), resume(), ShutDown(),Send(),SetDestinationID() , initQueues()

1. **What is the relation between Claim, Exposure, and Incident?**
2. **What are the Claim Validation Levels?**



1. **What are the Guidewire Upgrade Steps?**

Upgrade Steps

1. Create a Configuration Backup
2. Update Infrastructure
3. Install and Test GW 9 Base
4. Compare and Merge custom configuration changes with GW9
5. Compare and Merge Data Model Changes
6. Compare and Merge Rules
7. Validate and Resolve Errors in Configuration
8. Export Admin Data
9. Purge unused /address correction/completed workflows and logs
10. Run DB Validations
11. Create a Database Backup
12. Prepare DB for upgrade with all pre-requisites
13. Implement Data Model Upgrade Plugin to perform DB operations during upgrade
14. Disable Scheduler
15. Suspend Message Destinations
16. Configure DB for upgrade
17. Start Server and Perform auto DB Upgrade
18. Run DB Validations
19. Back up the upgraded DB
20. Validate Integrations and Configurations
21. **What is post on change?**
22. 
23. **What is Reflection?**

If an element can have a Reflect child, Studio displays a **Reflection** properties tab. The **Reflection** tab

has a checkbox for **Enable client reflection**, which indicates whether that element has a Reflect child.

If you enable client reflection, the **Reflection** tab also contains a properties list for the Reflect element

and a child list for its ReflectCondition children

1. **What can we assign to others at claim level?**

Claim, exposure, Activity, Matter

1. **What do we have in Activity patterns?**

Activity pattern code, subject, priority

To find (retrieve) a specific activity pattern, use one of the following Gosu find or get methods. The find

method returns a query object and the get method returns an ActivityPattern object.

ActivityPattern.finder.findActivityPatternsByCode("activity\_pattern\_code")

ActivityPattern.finder.getActivityPatternByCode("activity\_pattern\_code")

1. **What tabs available in Administration tab?**
2. **What are the other elements available in Financials other than Check like reserve, recovery?**
3. **How can you consume web service?**
4. **How can you configure Environment levels in Web service?**

You can configure environments by directly in .wsc(web service collection) itself.

1. **What steps will be called after Creating the Claim?**
2. **What are the attributes available in PCF?**

Id,post on change(),visible , required, editable,label,action,value,value range.

1. **Name some coverages in Commerical Auto or Personal Auto?**

Auto Physical Damage, Collision,Bodily Injury, Auto Liability, Property Damage, Uninsured Motorist, Underinsured Motorist, Med Pay

1. **Do you know Salvage? What is Salvage?**

If a vehicle has been completely destroyed or damaged and the insurer decides it is not worth fixing the damage, it is considered to salvage. After reimbursing the claimant for the value of the vehicle, the insurer can take possession of the vehicle and rebuild it or attempt to sell it for its parts. In Claim Center, you can assign an exposure to salvage by marking it as a total loss. You can then enter and manage salvage-related activities in the **Salvage** screens. Currently, salvage is used only for vehicle damage exposures and the Auto line of business.

1. **What is Gosu Enhancement?**

* A **Gosu enhancement** is a set of code that enhances the functionality of an existing Guidewire type
* For configuration developers, nearly all enhancement work involves enhancing either:
  + Guidewire entities
  + Java classes
* Enhancements can have any number of the following:
  + Getter properties, which calculate derived values
  + Setter properties, which set field values in situations where additional logic is needed
  + Methods, which:
    - Take input parameters, and/or
    - Modify other objects, and/or
    - Create objects

1. **MODES?**

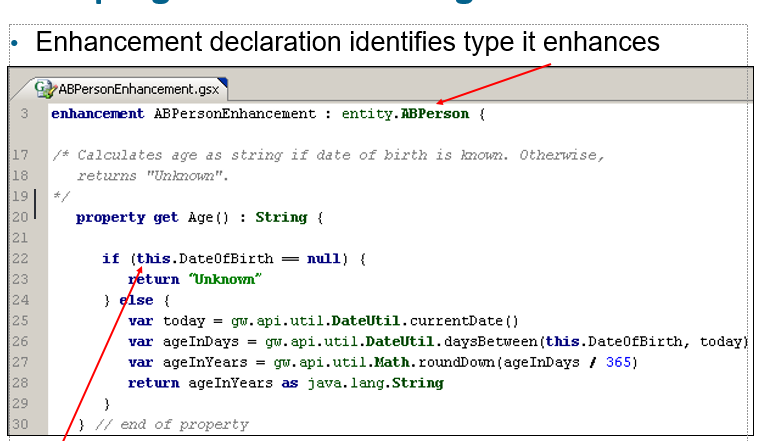
* **Mode** is a property used for PCFs that are appropriate for a given business scenario
  + Mode identifies which scenario(s) the PCF is for
  + Modal PCFs always come in sets
  + **Default Mode**; Modal PCF sets include one PCF with mode of "default", which is used when something references a mode that doesn't otherwise exist
* Modes can be used with:
  + Detail views
  + Input sets
  + List views
  + Card views
  + Info bars
  + Screens
  + Menu item sets
  + ...and more (refer to documentation)
* Modes are often used to accommodate variations in:
  + Contact type (individuals, companies, and so on)
  + Line of business (auto, workers' comp, property, and so on)

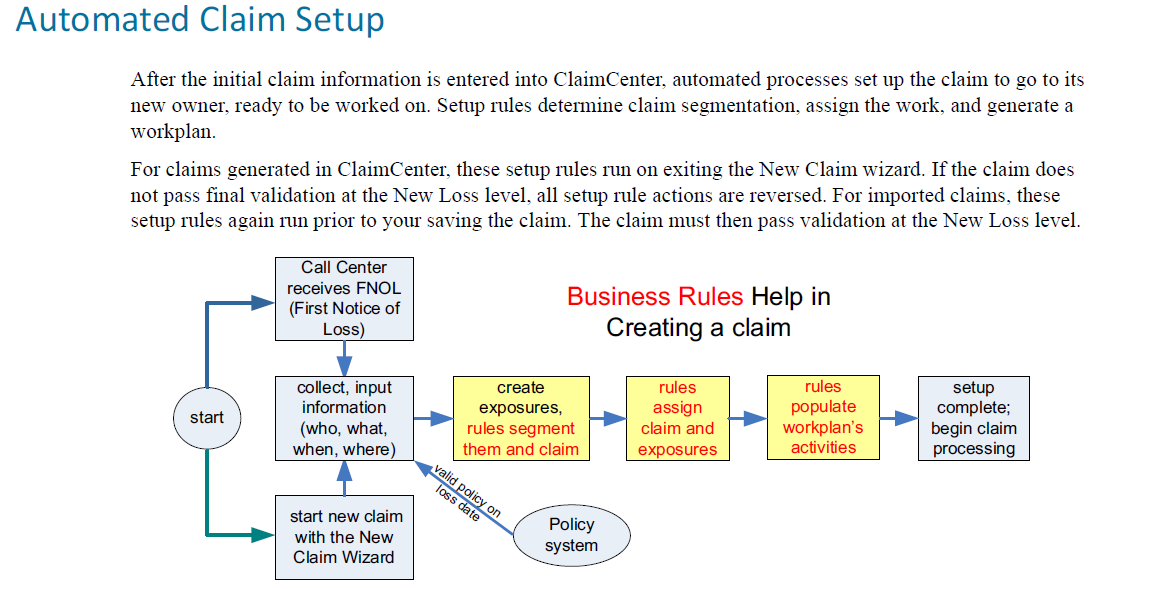
1. **What is Entity Builder?**

* An entity builder is a Gosu class used to create Guidewire objects and set the properties on those objects
  + Entity Builders are extensions of the platform class gw.api.databuilders.DataBuilder
  + Entity Builders can be extended by client written Gosu classes
  + Most objects in Guidewire Applications have a corresponding entity builder class
  + Entity builders handle required fields and required relationships
  + Can be used to create data objects for use in unit testing
  + Each test should:
  + Instantiate its own data
  + Manipulate the data
  + Verify the data
  + Functions for commonly created data can be written separately and called from different tests

1. Added DOB for user and how to display exact age for the person?

* A **getter property** is used to calculate a derived value
  + Property not declared at data model level
  + Value not stored in database
  + Code cannot receive input parameters
  + Code can only return value
  + Should not be used to alter any data
  + Null safe
* Example
  + ABPerson.Age
    - Derive value by calculating number of years between date of birth and current date (or returning "Unknown" if date of birth is null)
    - Storing value in database would be redundant





Claim Segmentation

Segmentation enables you to categorize incoming claims and their exposures into both segments and strategies

based on business criteria, such as:

• Segments that describe the type and severity of losses, such as multi-car, single car, injuries, and glass

only

• Segments that describe the loss location, such as close to field office

• Strategies based on policyholder type, such as normal, preferred, sensitive, or questionable

These category pairings, called segmentation, help assignment rules make good choices when deciding how to

handle the loss. For example, if the segment is Theft and the strategy is Preferred, then assign the claim to the

closest office for fast-track processing

Incidents, Exposures, and Claims

In working with incidents, claims, and exposures, it is important to understand the following:

• **Incidents** capture the information about what was lost, hurt, or damaged. They do not capture coverage,

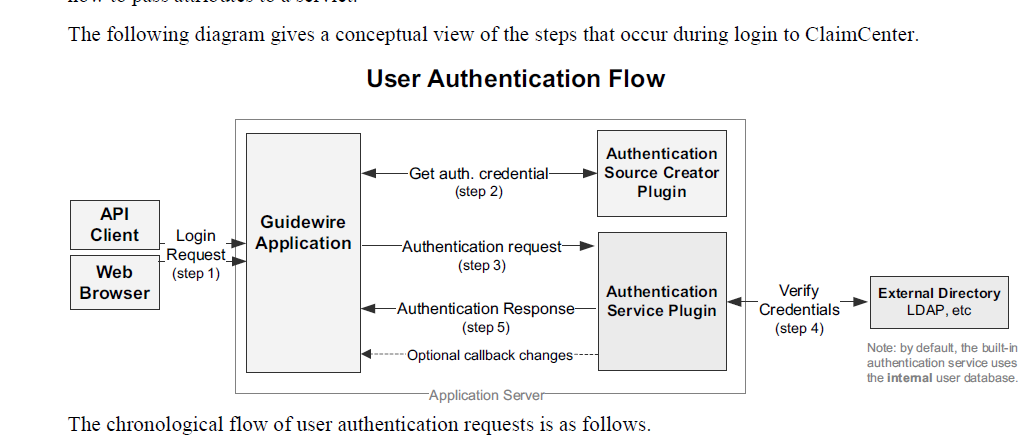
coding, financials, or other carrier involvement.

• **Incidents** and claims have a ***many-to-one***relationship—a single claim can have multiple associated incidents.

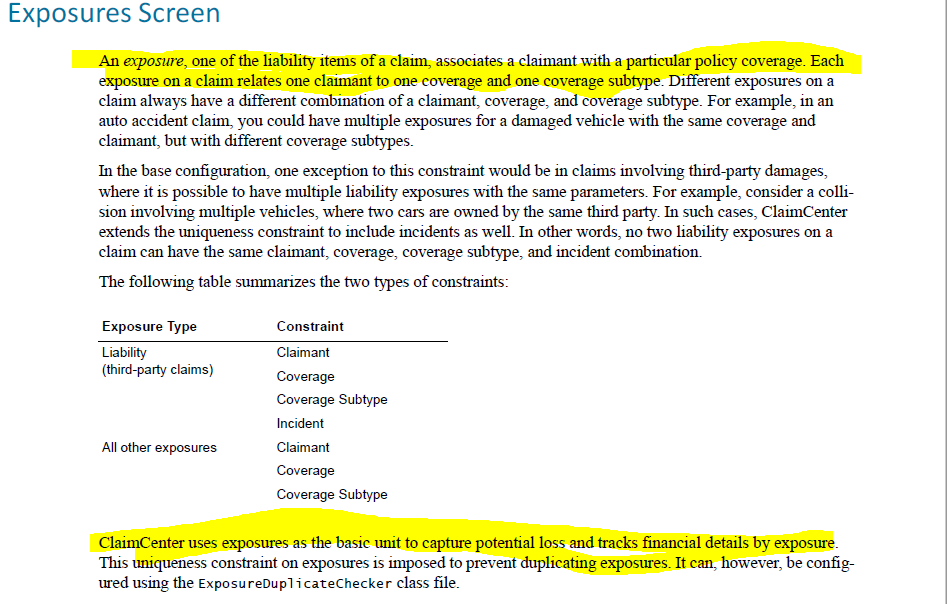
• **Incidents** and exposures have a ***one-to-many***relationship—a single incident can have multiple associated

**exposures**.

**User Authentication Integration**



**What is an exposure?**



**Regulatory Reporting?**

Berkley Risk Administrators is implementing a new claims system, Guidewire Claims Central for their P&C Lines of business. In order to adhere to regulatory compliance and avoid penalties, BRAC submits claim information to regulatory organizations such as CMS and ISO Statistical Reporting.

BTS is responsible for such submissions for all Alternative market companies. As one of the participating companies, BRAC extracts the claim information from CC conforming to a schema provided by BTS, creates the XML file and copies to a secure FTP location on a Berkley datacenter server. This XML file is consumed by BTS Statistical Data Reporting System to process as needed and submit to the different regulatory agencies.

1. **Define the broad purpose for each element category:**
   1. **Atomic widgets: Atomic widgets display individual elements of data and/or functionality.**
   2. **Container widgets: Container widgets group atomic widgets into logical groups.**
   3. **Locations: Locations define how users move from one place in the application to the next.**
2. **What does a PCF file define?**

**A PCF file defines a container widget or location and its contents**

1. **What does each PCF editor convention mean?**
   1. **A container widget shaded light blue**

**The container widget is referenced by the current PCF file. (Its contents are contained in another file**

* 1. **A container widget shaded dark blue**
  2. **A light green line**
  3. **A dark green line**
  4. **A red background**

1. **How do you deploy PCF changes? : ALT + SHIFT + L.**

Blocks are used in many situations where a method requires an expression as an input parameter, such as:

* + Arrays
    - hasMatch() - used to determine if any element in an array matches a given condition

**hasMatch( \ row -> row.LastName == "Andrews") would return true**

* + - firstWhere() - used to retrieve the first element in an array that matches a given condition

**firstWhere( \ row -> row.LastName == "Andrews") would return the third element (Samantha Andrews)**

* + - where() - used to generate a target array that consists of all the members of a source array that match a given condition
  + **where( \ row -> row.LastName == "Andrews") would return an array containing the third element (Samantha Andrews)**
    - These methods are discussed in greater detail in the next set of slides.
  + Queries
    - or() - used by queries objects to join multiple conditions together with OR logic
    - orderBy() - used by result set objects to specify logic for how to order the results in the set (in ascending order)
    - orderByDescending() - used by result set objects to specify logic for how to order the results in the set (in descending order)
    - These methods are discussed in greater detail in the "Queries" lesson of this course
  + Database transactions and bundles
    - runwithNewBundle() - used to create a bundle of run-time objects and then commit data in those objects to the database
    - This method is discussed in greater detail in the Application Integration courses