Information Technology Exception Request Management System

Ryan Uhl
CPET 491
Spring 2012

ITE Overview

- Background and Application Concept
- Architecture
- Application Pages
- Lessons Learned
- Demonstration

Background and Application Concept

ITE Interested Parties

- Corporate Standards and Architecture
- Information Security Risk Management
- Endpoint Management
- Andrew Dahman
  - ITT Exelis
    - Supervisor – Enterprise Applications
- Ryan Uhl
  - ITT Exelis
    - Project Manager and Application Developer
- Hongli Luo
  - IPFW
    - Faculty Advisor

4

ITE Summary

- ITT Exelis
  - Global Defense Contractor
  - Approximately 15000 users
- Current Process
  - Completed with Excel spreadsheet
    - User must know who to email to get approval
    - No standard path of approval
    - Changes take time to replicate through all users
  - Leads to confusion
    - Denials in one division and approval in another


5

ITE Summary (CONT)

- Centralize and standardize process
  - Requests
  - Approvals
- Web-Based Application
  - Database Driven
  - Real-time, On-Demand Reporting
  - Generate Email Notifications when request status changes
- Access
    - All network users (with credentials)
      - Inside and Outside company


6

ITE Scope

- Collect Requests
  - Hold requests until actioned (approved/denied)
- Notify approvers of pending requests by email
  - K2
- Provide common approval process
  - Track History
- Real-time reporting
  - Business Objects

7

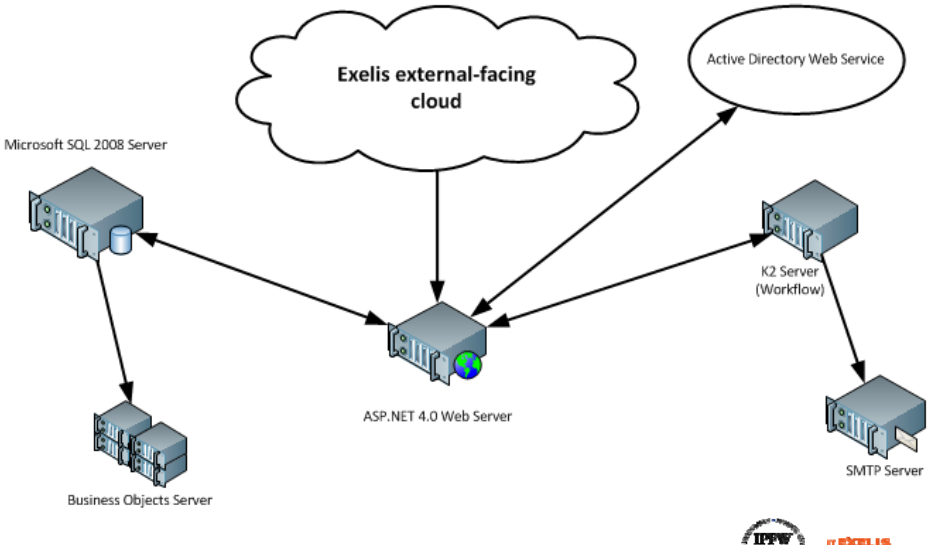
ITE Technologies

- Application written in C#
  - .NET Managed Language
- Microsoft SQL 2008
- IIS 7
- K2
  - K2 is also written in .NET
    - Allows for extendability and easy integration in application
  - Hosted in Exelis network on own WebServer/Database
- Business Objects
  - Java
    - Apache

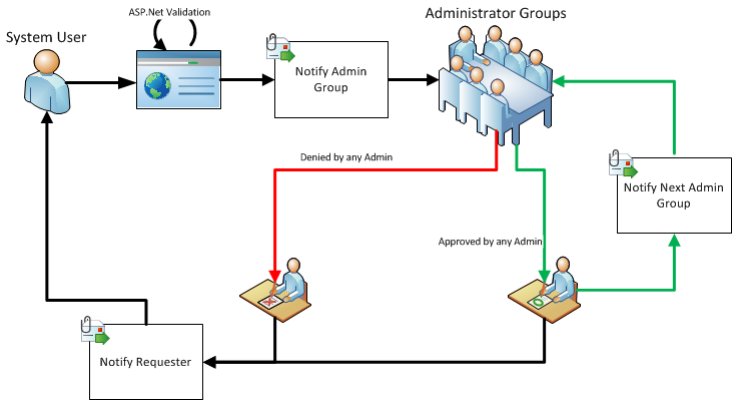
8

Architecture

ITE System Architecture

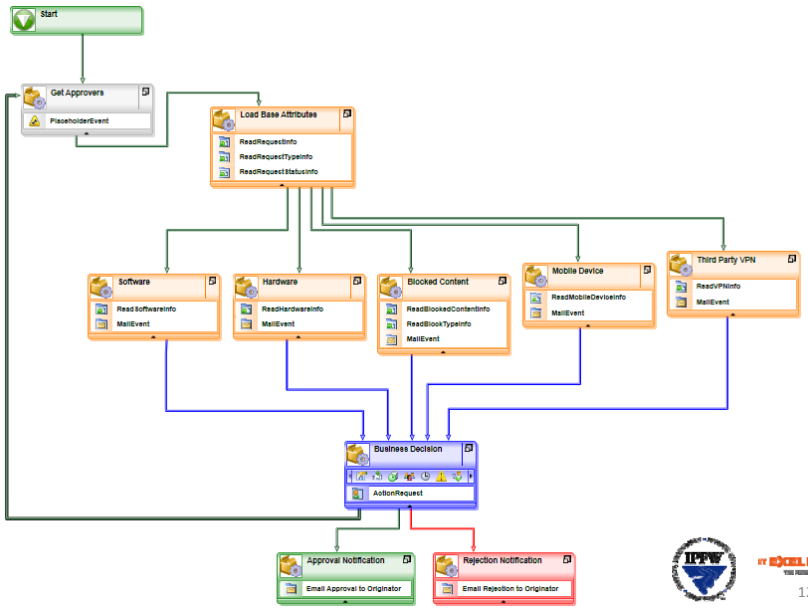


ITE User Behavior Architecture



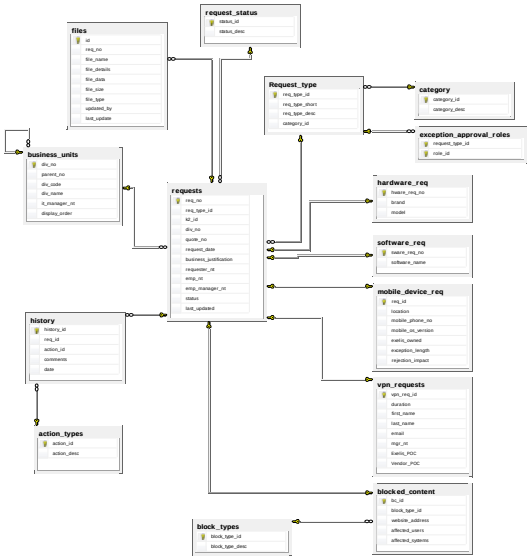
11

ITE Approval Workflow (K2)



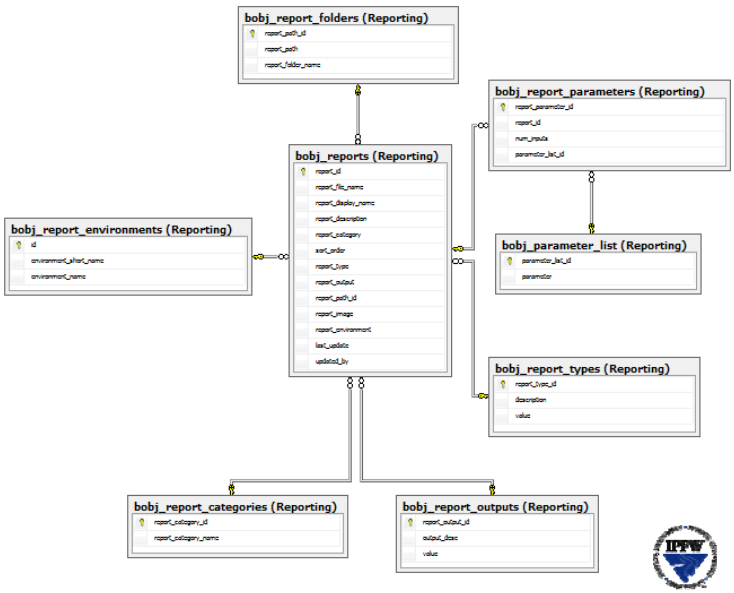
12

ITE Database Implementation (Request)



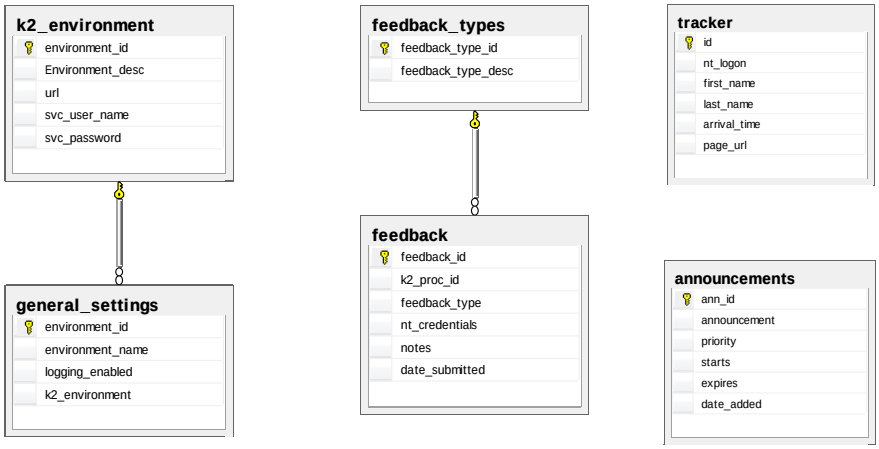
13

ITE Database Implementation (Reporting)

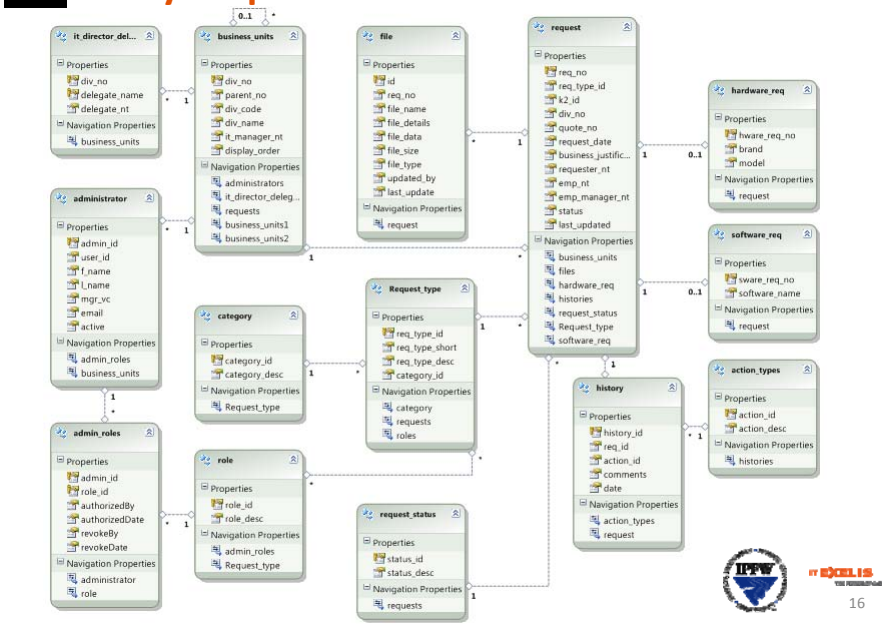


14

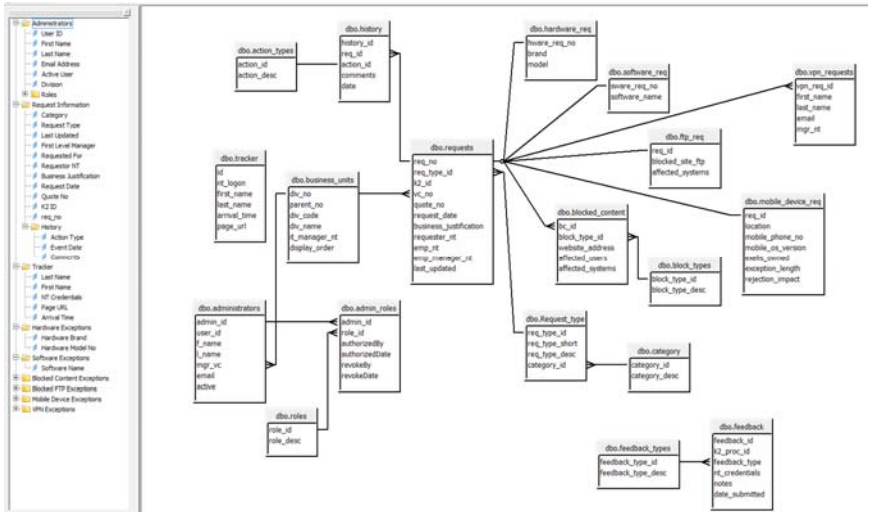
ITE Database Implementation



ITE Entity Request Model

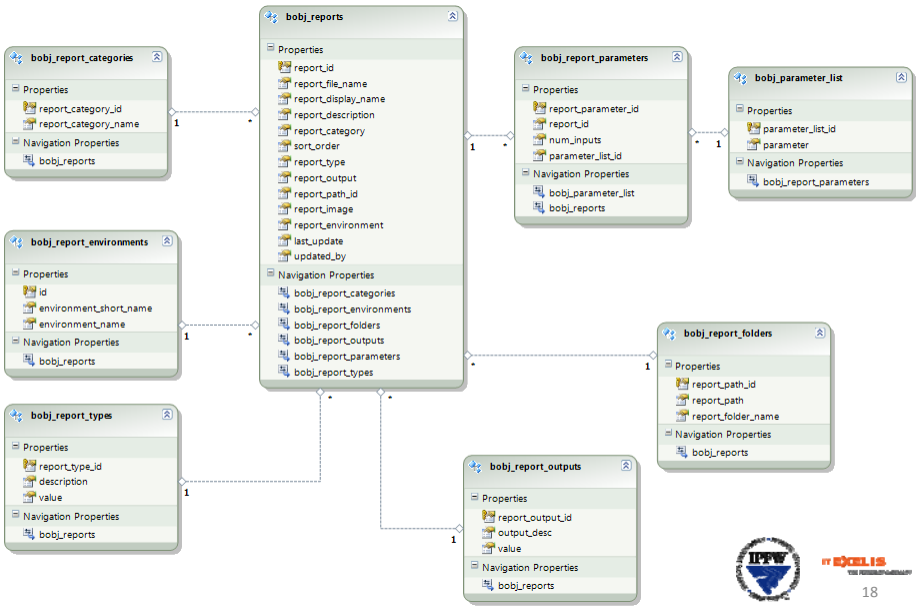


ITE Business Objects Universe



17

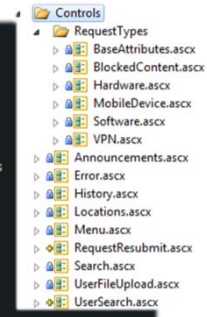
ITE Entity Reporting Model



18

ITE LINQ

```
//Save to the database
using (RequestEntity it = new RequestEntity())
{
    //Add the history element to state that the item was created
    history hist = new history();
    hist.comments = "System Created History";
    hist.req_id = req.req_no;
    hist.date = req.request_date;

    IQueryable<Models.action_types> at = it.action_types.Where(a => a.action_desc == "Create");
    hist.action_id = at.First().action_id;
    req.histories.Add(hist);

    //add the uploaded files to the request object
    List<IT_Exceptions.Models.file> files = fileUpload.getFiles();

    foreach (IT_Exceptions.Models.file file in files)
    {
        if (file.file_size > 0)
        {
            req.files.Add(file);
        }//end if
    }//end for

    //set the status of the request to open
    Models.request_status stat = it.request_status.Where(rs => rs.status_desc.Equals("Open")).FirstOrDefault();
    req.status = stat.status_id;

    req.req_type_id = Request_type.getRequestType(Request.QueryString["type"].ToLower());

    //save the request to the DB
    it.requests.AddObject(req);
    it.SaveChanges();
```



19

Application Pages

ITE Third Party Tools

- jQuery
  - Extends JavaScript Library to abstract mundane tasks
- jQueryUI
  - Provides User Interface that is consistent across browsers
- jQuery Tools
  - Tooltip replacement
    - Consistent look
- zoomImage
  - Click image to zoom to foreground
- DefaultText

21

ITE Third Party Tools

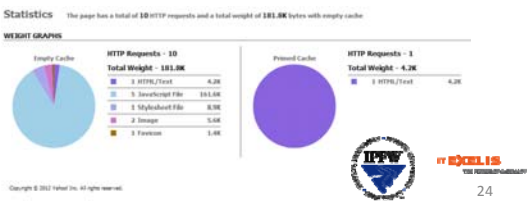
- SquishIt
  - Minify and compress JavaScript and CSS
- SmushIt
  - Online image sampling tool
    - Reduced image size while maintaining quality
- Sprites
  - Multiple images in one file
    - Reduces web server calls on page load
    - Reduces file size of needed images

22

ITE Sprites

23

ITE Homepage



24

ITE Risks

- Four points of failure
  - Mitigate
    - Ensured efficient connections
- No Entity Framework experience
  - Mitigate
    - Learn, research, evaluate
- Scope Creep is not managed
  - Accept
    - Almost impossible to avoid

25

ITE Risks

- Uncommon hierarchy/inheritance in entity (request to subtype)
  - Accept
    - Online research for best practice
- Mobile Interface Integration
  - Transfer
    - Customers never decided on mobile platform
      - Decided to use WP7 Mobile Internet Explorer

26

ITE Schedule

- Developer Cost
  - Estimated
    - 250 hours
  - Charged as of 4/20
    - 317 hours

27

Lessons Learned

ITE Lessons Learned

- Entity Framework
  - Great platform
    - Abstracts database interaction from developer
- SourceCode K2 BlackPearl
  - Refrain from using service brokers
- Project Management
  - Scope not managed correctly
  - Difficult to make senior groups come to consensus
- Application Design
  - Would have liked to use MVC 3

29

ITE Questions?

30