

Platea Integración Ficheros

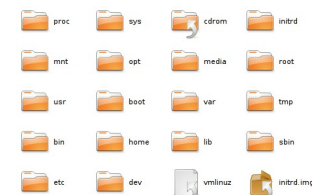
Difusión Otoño 2013

Soporte

Oscar Guadilla

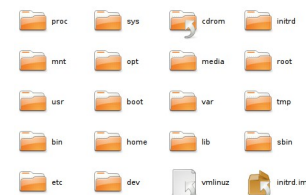


Índice



1. Requisitos
2. La Solución
3. Guías de Uso
4. Utilidades
5. PIF y EjieBOX

1. Requisitos

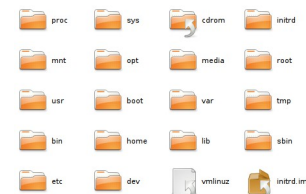


- funcionales
 - tanto para backend como para frontend
 - cualquier tecnología (java, .net...)
 - usable tanto para intercambios internos como externos (xc001)
 - total trazabilidad para saber que está pasando con los ficheros
 - usabilidad (gmail)
- técnicos
 - escalable & robusto
 - auditable desde el punto de vista de seguridad
 - estanqueidad en los datos (lopd)

- único → ubicado en integración
- sólo para intercambio → no mas de 15 días
- filesystem distribuido → hadoop (hdfs), homónimo al gfs de google
- virtual → los ficheros en /z80/ no se ven con un ls
- zonas por aplicación

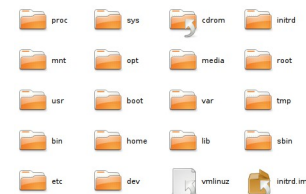
- básicas: put, get, move, copy, list, delete
- extras para api batch a nivel de directorio:
 - pud, getd, deleted
 - putdf, getdf, deletedf

2. La Solución (II)



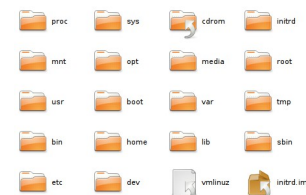
- canales
 - java: weblogic 8.1 y weblogic 11
 - rest y webdav
 - comandos batch, como n38bd
 - componente de cliente asíncrono (sesión usuario xlnets)
 - NOTA: no es una interfaz de usuario final es simplemente un botón y un cajetín
- seguridad
 - basada en xlnets: sesión de usuario o aplicación
 - basada en u/p: para aplicaciones que representan a terceros
 - Tipos
 - básica, no requiere configuración y se comporta como una apartado de correos
 - avanzada, requiere configuración (acls) para escritores: aplicaciones o usuarios

2. La Solución (III)



- eventos
 - cada vez que se escribe un fichero → WRITE
 - suscripciones con ruta PIF
- trazabilidad
 - consola pública en todos los entornos
 - información de auditoria en las trazas:
 - quien
 - cuando
 - donde
- expurgos
 - procesos nocturnos en base al ttl suministrado en los comandos put
 - se hace backup diario previo al expurgo de las zonas de ficheros

3. Guías de Uso (I)



- Normativa de Desarrollo publicada en el portal del Desarrollador
- Si se usa modelo básico de seguridad no es necesaria ninguna configuración extra
- Ejemplos
 - comandos

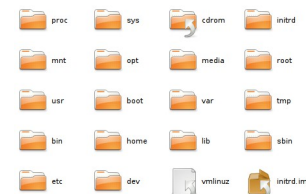
```
/aplic/y31/cade/y31ApiBatch.sh t17i list /t17i
```

```
/aplic/y31/cade/y31ApiBatch.sh t17i put /tmp/prueba.txt /t17i/prueba.txt true 11111
```

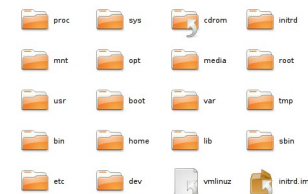

3. Guías de Uso (II)

– código java

```
// obteniendo un objeto service
Y31JanoService service=Y31JanoServiceAbstractFactory.getInstance();
// Preparando los parámetros
Document sesion=null;
String from="/r09/";
// invocando el metodo
List files=service.list(sesion, from);
Y31AttachmentBean resul=null;
for (int i=0; i<files.size(); i++) {
    resul=(Y31AttachmentBean)files.get(i);
    System.out.println("-->" +resul.getContentType());
    System.out.println("-->" +resul.getFileName());
    System.out.println("-->" +resul.getFilePath());
    System.out.println("-->" +resul.getSize());
}
```



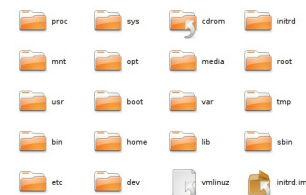
3. Guías de Uso (III)



– componente de cliente

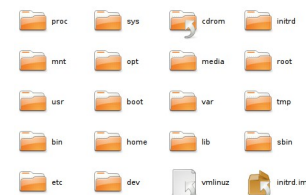
```
function inicio() {  
    var settings = {  
        base_url : "http://svc.integracion.jakina.ejiedes.net",  
        folderPath: "/r02",  
        custom_settings : {  
            language: "es",  
            progressTarget : "fsUploadProgress"  
        },  
        button_placeholder_id : "spanButtonPlaceholder",  
        button_width: 75,  
        button_height: 22,  
        file_size_limit: "10 MB",  
        file_types: "*.rar",  
        file_types_description: "selec. fichs.",  
        upload_success_handler : uploadSuccess  
    };  
    var widget = new JANO.Uploader(settings).create();  
}  
function uploadSuccess(file, serverData) {  
    var progress = new FileProgress(file, this.customSettings.progressTarget);  
    progress.setComplete();  
    progress.setStatus("Completado");  
    progress.toggleCancel(false);  
    var ooo = serverData.evalJSON(true);  
    filesCount = filesCount + 1;  
    ...  
    currentFile.setAttribute("value", ooo.filePath);  
    ...  
}
```

3. Guías de Uso (IV)



- ejemplos de uso reales en producción
 - educación (componente cliente, api java, comandos)
 - editran ip (comandos). pif homónimo al uso de /datos/edi
 - SIPCA (gtt)
 - Asuntos Sociales (ayuntamiento de bilbao)
 - Industria + Notificaciones Servicio Correos (akropost)
 - datos de laboratorios de osakidetza
 - Dokusi y Tramitación están ofreciendo interfaces PIF
- posibles ejemplos de uso
 - Servicios Horizontal de Firma. paso de ficheros por referencia
 - *Dokusi. paso de ficheros por referencia*
 - *Visión Ciudadano. aportación de documentos*
 - /datos/interc

4. Utilidades



Platea Integracion Ficheros

Consolas: [Desarrollo](#) , [Pruebas](#) , [Produccion](#)

Explorador de Ficheros: [Desarrollo](#) , [Pruebas](#) , [Produccion](#)

EjieBOX: [Desarrollo](#) , [Pruebas](#) , [Produccion](#)

Manager: [Desarrollo](#) , [Pruebas](#) , [Produccion](#)

Librerias: [Weblogic 11, Librerias, Config y Ejemplo](#) , [Weblogic 8.1, Librerias, Config y Ejemplo](#)

Kit de Terceros: [Kit de Terceros basado en API Rest y curl](#)

Documentos: [Normativa de Desarrollo](#)

Traza de operaciones

Ver todos los ficheros. SOLO DESA Y PRU

Intercambio de Ficheros
grandes entre personas

Intercambio de Ficheros con Explotación.
P.ej. cargas de CDs

5. PIF y EjieBox



zonas apps



interfaz técnica



interfaz técnica



zonas personas



interfaz web



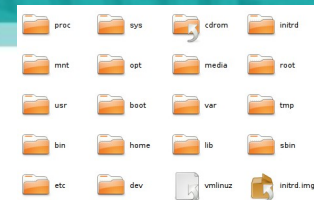
interfaz técnica

Uso Aplicaciones de osakidetza, interior, eustat...

Uso Aplicaciones Departamentales en EJIE

Uso Personas en Navegador Web

Uso Personas en Móvil o en Puesto (Z:)



¿dudas?

