

WORKFLOW

ESETTANULMÁNYOK

Viczián István

IP Systems

<http://jtechlog.blogspot.hu/>

JUM XIX. - 2012. szeptember 18.

- ▶ Két projekt
- ▶ Mindkettőben folyamatirányítás
- ▶ Eltérő követelmények
- ▶ Eltérő megoldások

- ▶ Dokumentum gyártási folyamat
- ▶ Üzemeltetés közeli
- ▶ Létező Windows batch script-ek, C, Java alkalmazások
- ▶ Célok
 - Időmérés
 - Naplózás
 - Monitorozás

- ▶ Apache Ant
- ▶ Platformfüggetlen
- ▶ XML formátum
- ▶ Project, target, task
- ▶ Deklaratív, nem imperatív, futási sorrend függőségek alapján
- ▶ Külön futtatható target-ek
- ▶ Eseménykezelők
 - Egységes időmérés, naplózás, monitoring, egymásra futás

- ▶ Feladat (job): egy konkrét gyártási folyamat a hozzá tartozó munkafolyamattal és beállításokkal. Technikailag egy build.xml és hozzá tartozó build.properties. Tulajdonságai ezen kívül a tetszőlegesen megadott név és leírás, valamint a legutolsó futás ideje, státusza, és a legutolsó sikeres futás ideje és hossza.
- ▶ Futás (run): egy feladat konkrét lefutása, státusszal (sikeres/sikertelen), kezdési időponttal, futási hosszal, keletkezett napló állománnyal
- ▶ Lépés (target): a feladaton belüli csoportosítás, egyes lépések. Technikailag egy Ant target-nek felel meg. Ennek tulajdonságai: név (target name attribútum), leírás (target description attribútum), kezdési időpontja, futási hossza.

- ▶ Listener
- ▶ Log4J
- ▶ Külön napló állományba lehessen gyűjteni a futási időket
- ▶ A különböző target-ek külön napló állományba naplózzanak
- ▶ A napló állományok nevében benne legyen a futtatás pontos időpontja
- ▶ A napló állományokat akár naponta külön könyvtárba tegye a naplózó rendszer

- ▶ Cron-nal ütemezhető, lehetne Java-ból is
- ▶ Property
- ▶ Futtatandó lépés kiválasztása
- ▶ Ütemezett folyamatok egymásra futása
- ▶ Több build.xml állomány
- ▶ Dokumentációs task: az implementáció folyamán a dokumentálás megkönnyítésére build folyamatot vezérlő konfigurációs állományban elhelyeztünk dokumentációs megjegyzéseket, melyből később PDF, HTML, stb. formátumú dokumentáció gyártható XSL-FO technológia segítségével
- ▶ Saját Ant task (TCP kommunikáció)
- ▶ Verziókezelés: Subversion

- ▶ Üzemeltetés közeli
- ▶ Iteratív szemlélet hiánya: Contrib

- ▶ Alkalmazás modul
- ▶ Meglévő képernyőkhöz megjelenítésben illeszkedni
- ▶ Bonyolult munkafolyamat
- ▶ Különböző szereplők
- ▶ E-mail értesítések, timeout-ok
- ▶ Vezetői áttekintés
- ▶ Lehetőleg Spring-hez integrált

- ▶ Előzmény: JBoss jBPM
- ▶ Activiti (Alfresco)
- ▶ Apache licence
- ▶ Activiti Modeler
- ▶ Activiti Designer - WYSIWYG Eclipse alapú workflow designer
- ▶ Activiti Explorer
- ▶ REST API
- ▶ Finom jogosultságkezelés

- ▶ Az OMG (Object Management Group) által 2004 óta fejlesztett
- ▶ Grafikus & XML
- ▶ Grafikus megjelenítésre vonatkozó információk
- ▶ Gyártófüggő kiegészítések

- ▶ Process
- ▶ Event
 - Throwing, catching
 - Start, Signal, Message, None end, Error end, Cancel end, Boundary, Timer
- ▶ Activity (Task)
 - User task, script Task, Java Service Task, Web Service Task, Business Rule Task, Email Task, Mule Task, Manual Task, Java Receive Task, Shell Task, Execution Listener, Task Listener
- ▶ Sequence flow
- ▶ Gateway
 - Exclusive, Parallel, Inclusive, Event-based
- ▶ Execution

- ▶ Hierarchikus modellezés
- ▶ Esemény hatóköre

- ▶ Hamar látványos megoldás
- ▶ Remek dokumentáció
- ▶ Jó támogatás
- ▶ Pörgő fórum, JIRA
- ▶ Maximális Spring integráció
- ▶ Könnyen unit tesztelhető, embedded database
- ▶ Workflow definition verziókezelés
- ▶ Tranzakciókezelés: egyszerűen működik
- ▶ Activiti in Action

- ▶ Problémák: workflow migráció (workflow, tárolt paraméterek, hívott szolgáltatások, listener-ek)
- ▶ Nem támogatott BPMN 2.0
- ▶ Főleg task alapú interfész
- ▶ Munkásabb history kezelés
- ▶ Fogalomkeveredés, változó API

- ▶ Teljesen más gondolkodásmód
- ▶ Jól jön felmérésnél is
- ▶ Limitált
- ▶ Nem érdemes „mögé nyúlni”
- ▶ Kööttség az ügyfél oldalán is
- ▶ Nincsenek elvarratlan szálak
- ▶ Mindig up-to-date modell, dokumentáció
- ▶ Látványos
- ▶ Képernyő generálás?
- ▶ Wrapper