

# Convert standard mail notifications to use the Mail-Ext Publisher plugin

This script replace mail notifications in all projects by Mail-Ext publisher plugin and re-uses existing recipients.  
The Mail-Ext is configured with :

- Recipient list : current recipient list defined in the project (the MailExt isn't activated if the project doesn't have yet a mail notification)
- Mail subject : default subject from global config
- Mail body : default body from global config
- Triggers : Failure, StillFailing, Unstable, StillUnstable, Fixed triggers are activated and send mails to the recipient list

It was tested on Freestyle and Maven projects with Hudson 1.346.

```
import hudson.plugins.email.*
import hudson.plugins.email.plugins.trigger.*
import hudson.model.*
import hudson.maven.*
import hudson.maven.reporters.*
import hudson.tasks.*

// For each project
for(item in Hudson.instance.items) {
    def recipients
    println("JOB : "+item.name);

    // Find current recipients defined in project

    if(item instanceof MavenModuleSet) {
        println(">MAVEN MODULE SET");

        // Search for Maven Mailer Reporter
        println(">>Reporters");
        for(reporter in item.reporters) {
            println(">>> "+reporter);
            if(reporter instanceof MavenMailer) {
                // save recipients
                recipients = reporter.recipients
                // remove reporter
                item.reporters.remove(reporter)
            }
        }
    } else
    if(item instanceof FreeStyleProject) {
        println(">FREESTYLE PROJECT");
    }

    println(">>Publishers");
    for(publisher in item.publishersList) {
        println(">>> "+publisher);
        // Search for default Mailer Publisher (doesn't exist for Maven
projects)
        if(publisher instanceof Mailer) {
            // save recipients
            recipients = publisher.recipients
            // remove publisher
            item.publishersList.remove(publisher)
        } else
```

```

    // Or for Extended Email Publisher
    if(publisher instanceof ExtendedEmailPublisher) {
        // save recipients (if it already exist it overrides others
values)
        recipients = publisher.recipientList
        item.publishers.remove(publisher)
        // remove it
        item.publishersList.remove(publisher)
    }
}

// If we found recipients list to send mail
if(recipients!=null){
    println(">CURRENT RECIPIENT : "+recipients)
    // We create a new Extended Email Publisher
    def eep = new ExtendedEmailPublisher();
    eep.recipientList = recipients
    eep.defaultSubject = "\$DEFAULT_SUBJECT"
    eep.defaultContent = "\$DEFAULT_CONTENT"
    // With some triggers
    eep.configuredTriggers.add(new FailureTrigger(
        email : new EmailType(
            sendToRecipientList : true,
            sendToDevelopers : true,
            includeCulprits : true,
            body : ExtendedEmailPublisher.PROJECT_DEFAULT_BODY_TEXT,
            subject : ExtendedEmailPublisher.PROJECT_DEFAULT_SUBJECT_TEXT
        )))
    eep.configuredTriggers.add(new FixedTrigger(
        email : new EmailType(
            sendToRecipientList : true,
            sendToDevelopers : true,
            includeCulprits : true,
            body : ExtendedEmailPublisher.PROJECT_DEFAULT_BODY_TEXT,
            subject : ExtendedEmailPublisher.PROJECT_DEFAULT_SUBJECT_TEXT
        )))
    eep.configuredTriggers.add(new StillFailingTrigger(
        email : new EmailType(
            sendToRecipientList : true,
            sendToDevelopers : true,
            includeCulprits : true,
            body : ExtendedEmailPublisher.PROJECT_DEFAULT_BODY_TEXT,
            subject : ExtendedEmailPublisher.PROJECT_DEFAULT_SUBJECT_TEXT
        )))
    eep.configuredTriggers.add(new StillUnstableTrigger(
        email : new EmailType(
            sendToRecipientList : true,
            sendToDevelopers : true,
            includeCulprits : true,
            body : ExtendedEmailPublisher.PROJECT_DEFAULT_BODY_TEXT,
            subject : ExtendedEmailPublisher.PROJECT_DEFAULT_SUBJECT_TEXT
        )))
    eep.configuredTriggers.add(new UnstableTrigger(
        email : new EmailType(
            sendToRecipientList : true,
            sendToDevelopers : true,
            includeCulprits : true,

```

```
        body : ExtendedEmailPublisher.PROJECT_DEFAULT_BODY_TEXT,  
        subject : ExtendedEmailPublisher.PROJECT_DEFAULT_SUBJECT_TEXT  
    )))  
    // And we add/replace it in the project  
    item.publishersList.replace(eep);  
}else{  
    println (>NO RECIPIENT")  
}
```

```
println("\n=====\n");  
}
```