Automated Build Generator - Bug #2520
dependencies reported to jenkins for CI

03/14/2016 03:16 PM - R. Haschke

Status:	Resolved		**Start date:**	03/14/2016
Priority:	Urgent		**Due date:**	
Assignee:	J. Moringen		**% Done:**	100%
Category:			**Estimated time:**	0.00 hour
Target version:	0.6			

Description

Die Liste der nachgelagerten Projekte im jenkins ist bei vielen Projekte zu lang.

Nehmen wir mal drei Projekte mit folgenden, beispielhaften Abhängigkeiten:

A

B depends on A

C depends on A,B

In diesem Fall sollte Projekte C nur Projekt B "watchen", weil die dependency auf A automatisch durch ein rebuild von dependency B erfüllt wird.

Da der build-generator potentiel erstmal sehr viele Abhängigkeiten findet, wäre ein solches pruning der Abhängigkeiten in jenkins m.E. auf jeden Fall notwendig.

Zur Zeit triggert ein rebuild von A:

rebuild B, C

rebuild C (after B)

Bei einen großen Projektbaum skaliert das exponentiell schlechter. Bei einer Abhängigkeitsliste A->B->C->D->E->F würde ein rebuild von A insgesamt 5+4+3+2+1 = 15 weitere builds triggern - obwohl nur 5 nötig wären.

Die Lösung müsste einfach zu implementieren sein:

Man nehme 1. die Abhängigkeitsliste des aktuellen Jobs (wie Du sie schon hast)

und subtrahiere 2. die Abhängigkeitslisten der dort gelisteten Jobs.

Also für unser Beispiel:

A: {} - {} = {}

B: {A} - {} = {A}

C: {A, B} - {} - {A} = {B}

Associated revisions

Revision 9e6d2260 - 04/08/2016 01:06 PM - J. Moringen

Added dependencies.mode behavior in src/project/classes-model.lisp

fixes #2520

Consider the following dependency structure

```
    A<---B<---C
    ^       |
    `--------'
```

The following values are allowed:

    direct (the default)

All jobs corresponding to direct dependencies trigger a given job.

In the above example, A and B trigger C.

minimal

Jobs corresponding to direct dependencies which are not among the
  dependency closure of the direct dependencies trigger a given jobs.

In the above example, only B triggers C (since A is in the
  dependency closure of the direct dependency B).

There is no triggering by upstream jobs.

- src/project/protocol.lisp (header): updated copyright
    (minimal-dependencies): new generic function; compute minimal
    dependencies as described above
    (minimal-dependencies t): new method; default behavior
- src/project/classes-model.lisp (header): updated copyright
    (deploy-dependencies job): depending on the value of the
    dependencies.mode variable, use direct, minimal or no dependencies
- src/project/package.lisp (header): updated copyright
    (package jenkins.project): added exported symbol minimal-dependencies

History

#1 - 04/08/2016 12:08 PM - J. Moringen

- Status changed from New to In Progress

- % Done changed from 0 to 20

#2 - 04/08/2016 01:09 PM - J. Moringen

- Status changed from In Progress to Resolved

- % Done changed from 20 to 100

Applied in changeset commit:9e6d2260811adc2e135bc47c82e2d3144305e6af.

#3 - 06/18/2016 02:16 AM - R. Haschke

Hand which option in distribution or template needs to be set for this feature?

#4 - 06/18/2016 01:44 PM - J. Moringen

R. Haschke wrote:

> *Hand which option in distribution or template needs to be set for this feature?*

The variable name is dependencies.mode. I recommend setting it via the -D commandline option. Possible values are:

- direct (the default)

All jobs corresponding to direct dependencies trigger a given job.

In the above example, A and B trigger C.

- minimal

Jobs corresponding to direct dependencies which are not among the dependency closure of the direct dependencies trigger a given jobs.

In the above example, only B triggers C (since A is in the dependency closure of the direct dependency B).

- none

There is no triggering by upstream jobs.

#5 - 09/30/2016 10:46 AM - J. Moringen

- Target version set to 0.6