To All:

This is a response to the research questions in the Hackystat JPL Build System: Overview and Initial Results document of 10/24/03.

**[Q.3.1.a] Do the 78 build entries correctly reflect the total number of builds that were undertaken between January and August?**

The builds reported in the initial data were incomplete. The scripts that extracted the build information were looking at a build area populated as each build occurred. Unfortunately, the build area was being periodically "cleaned up" by the MDS CM administrator to free up disk space.

However I have been able to capture this information from logs maintained by the CM administrator. The up-to-date build information from 1/03 to 12/03 is available for redelivery to Hackystat personnel.

**[Q.3.2.a] Should the Harvest Package Lifecycle documentation be augmented to discuss the additional demotion paths as practiced by the MDS development group and the circumstances under which they are used?**

The MDS Lifecycle definition should be updated to document these additional valid migration paths.

**[Q.3.2.b] Can CM Build demotion transition types and frequencies serve as evidence of positive/negative changes to the development process?**

The CM administrator will demote packages from CM Build to Build Queue state for the following reasons:

- problems in build just performed and additional package being developed to fix problem
- problem in build just performed and unable to locate associated developer
- dependency problem with other packages
- developer requests different package groupings
- CM administrator mistake

The CM administrator will demote packages from CM Build to Dev state for the following reason:

- problems in build just performed and additional package changes to be done by developer

**[Q.3.2.c] Under what circumstances are packages appropriately demoted from CM Build to Dev Complete?**

This transition is not necessary. It will be removed from the Harvest lifecycle.

**[Q.3.2.d] Why are there IAR packages without any corresponding state?**

The referenced IARs were only defined in the Build info command file. The Build info command file was extracted 12 hours after the Package info command file was generated. These 2 IARS were introduced within this window. Some filtering should be take place on either the Harvest extraction side on on the Hackystat side.

**[Q.3.2.e] Why can non-CP packages enter the workflow in states other than Created?**

The Package info command file was generated using package transitions beginning on 1/1/03. Any creations prior to this date are not included in the Package info command file. Hackystat should be adjusted to accommodate this. Note that "Created" is not a state, just a pseudo state to fit our syntax in the Package info command file.

**[Q.3.2.f] Why do certain packages exhibit an inconsistent set of State Change transitions?**

This is a bug in the scripts used to generate the Package info command file. I have fixed this problem.

**[Q.3.3.a] Why do large numbers of packages exhibit apparently "illegal" sequences of state transitions?**

Same answer as for Q.3.2.f.

**[Q.3.3.b] Under what circumstances is it appropriate for a package to transition from Dev to Dev Complete without any attached files?**

There is no requirement in our methodology that files be changed in order for a package to be closed. In some cases a fix associated with another package will also address the issues in this package. Also, a problem may be closed as non-reproducible.

As per our specification of the format of the Package info command file, files associated with a package will only be documented when the package moves from state Dev to Dev Complete. If a package is demoted back to Dev and then promoted to Dev Complete again, ONLY those files changed since the previous promotion to Dev Complete will be documented. Hackystat should be adding these new changes if any to the previous list of changed files.

**[Q.3.4.a] Does the cyclical rise and fall of promotions represent a predictable pattern of development activity?**

I will let someone else address this issue.

**[Q.3.4.b] Does the relatively constant level of demotions represent a predictable pattern of development activity?**

I will let someone else address this issue.

**[Q.3.4.c] Do we need to add a selector to choose the particular package types in the analysis?**

I think it would be very useful to be able to select based on package type. I would expect to see significant differences in promotion/demotion activity for CPs vs IARs.

**[Q.3.4.d] Will the current baseline levels of promotions and demotions change as new developers are added to the project or other process changes occur?**

I will let someone else address this issue.

**[Q.3.5.a] Are significant numbers of packages are left to "languish" in test complete for weeks at a time, followed by a "cleanup" phase every couple of months?**

The methodology is set up such that as each package passes the test suite, the package is moved to Test Complete. When a release is to be made, all the packages in Test Complete are moved to Release. For purposes of metrics, I believe that we can interpret that a package is “closed” when it moves to Test Complete.

**[Q.3.5.b] Is a cyclical change in the level of state change transitions accidental, deliberate, or emergent?**

I will let someone else address this issue.

**[Q.3.6.a] To what extent does this representation of work vs. rework based upon transition data accurately model the effort allocated by the development group to work vs. rework?**

I will let someone else address this issue.

**[Q.3.6.b] Does the decrease in percentage rework and unscheduled work indicate an actual process improvement?**

I will let someone else address this issue.

**[Q.3.6.c] Why are there no CPs with an associated IAR in our dataset?**
IARs are associated only with CPRs. IARs are not associated with CPs.

**[Q.3.6.d] Why do some IARs reference undefined CPRs?**

The was one CPR referenced by an IAR that does not exist. The CPR must have been deleted at some point in the development lifecycle. This should not happen.

**[Q.3.8.a] Can factors be identified that help predict the eventual age of a CP, IAR, and IM?**

I will let someone else address this issue.

**[Q.3.10.a] Does the apparent increasing trend in CP state age variation indicate development problems?**

I will let someone else address this issue.

I am ready to regenerate the the Package info command file and Build info command file whenever it is requested. Following the next extraction of data from 1/1/2003 to the current time, I will turn on my scripts that will extract any new package movement and builds since the last sample.

Rich Hug