Data-Driven Campaign Management at the National Ignition Facility

Presentation to
14th International Conference on Accelerator & Large Experimental Physics Control Systems (ICALEPCS)
October 6-11, 2013

Douglas Speck
Campaign Management Project Lead
What do we mean by “Campaign Management” and “Data Driven”

• “Campaign Management” refers to the software, data, and workflows through which we setup, review, approve and export experiments on the NIF

• Campaign Management Suite applications include:
  — Campaign Management Tool: experiment setup editor
  — Parts and Lists Manager: db app to manage CMT setup options
  — Approval Manager: manages approval workflow of completed setups
  — Shot Setup Reports: formats experiment setup selections into reports for review

• Within the context of Campaign Management, an “experiment” is an XML document that captures all of the shot parameters that experimenters specify
  — A “data group” is the set of parameters for one diagnostic, one target system, etc.

• “Data Driven” describes an application architecture in which parameters that are key determinants of the logical flow are stored and managed independently of the application logic.
  — The interface(s) to the data separately support access by the application and also by maintainers of the system.
Why use data-driven architectures?

- **Change is constant** – new diagnostics, updated diagnostics, new components, software feature updates

- **Software development is expensive**
  - Participation of multiple developers and full-time test engineer
  - Deployments needed to multiple development environments plus Production
  - Deployments to Prod interrupt production operations for several hours

At the cost of some increase in software complexity, “data-driven” lets us replace software release cycles with much cheaper data deployments
Campaign Management Tool (CMT)

- Spreadsheet-style experiment editor
- Sets pulse, energy, timing, pointing, and laser diagnostics configuration for 192-beam NIF laser, configurations for ~40 target chamber diagnostics, 2 target systems managed ~20000 parameters managed
Parts and Lists Manager (PLM) separates management of CMT setup options from CMT codebase

- Helps manage a major source of change – resources associated with target diagnostics such as filters, attenuators, detectors, cameras, delay lines, etc.
- With PLM, these changes can be performed when convenient, and asynchronously with respect to CMT code deployments
PLM flexibility carries a few challenges...

• Many part lists and the count is growing. List loads are a major component of CMT startup processing.
  • Recent re-design resulted in better than 2x speedup but this issue won’t go away...

• Stale data is created if a PLM entry is removed or modified after being selected in an experiment. This may cause validation errors & interrupt workflow on the experiment.
  • We’ve begun tracking all selections in experiments and know immediately which experiments become stale due to a change.
  • Future capabilities will include ability to push updates to CMT clients during editing sessions.

• With Production, Formal Test, Integration, and Development environments all managed separately, keeping data consistent is an ongoing nuisance.
  • Specially formatted Excel files read and written by PLM let us migrate arbitrary selections of table rows and columns between environments
Approval Manager

- Manages workflow for review, approval, export of experiments
- Data-driven by design
- Since 2011 introduction, approval workflow has been revised, expanded, diversified to enhance support of production operations
Approval Manager Lessons Learned: Data Group Dependencies

- After splitting the experiment into “data groups” to enable parallel review of unrelated setups, it turned out that they weren’t all unrelated
  - Solution: configurable data group dependencies + workflow logic that rescinds approval of dependent data groups if the independent parent data group loses its approval
  - A dependency template defines all possible dependencies; whenever an experiment is saved it’s configuration is scanned to determine if any parent/child dependency pairs exist and instantiates them for that experiment

Example of instantiated dependencies for an experiment
Approval Manager Lessons Learned: Dynamic Data Groups

• Some diagnostic instruments lack configurable support in CMT
  • Some diagnostic devices are “captives” of primary instruments, part of their hardware but not independently configurable
  • An instrument may not be planned for regular use at NIF so explicit CMT support is not warranted
• Enable review & approval for these by defining Dynamic Data Groups for them, then instantiating those DDG’s in experiments that need them
  • DDG definition includes a name, an approver role, and a default priority
  • Each DDG definition is saved permanently for reuse as needed
  • DDG may be created on an experiment by manual intervention or by programmed logic
Shot Setup Reports

• Generates electronic reports (HTML, Excel) for each data group to support review and approval

• Launched from report links in Approval Manager

• Three primary components
  • Report generating scripts stored as fields in a database
  • ShotSetupReports Java application – retrieves experiment XML and report script from database then executes script against XML
  • ShotSetupReports Admin page with report script editor

• Critical design aspect – storing report scripts as data
  • A typical change is to update an XPath for getting a parameter from experiment XML
  • Usually performed within minutes (if not seconds) of receiving request
  • System stays online throughout update cycle
  • Updated report is available as soon as updated script is committed to database
Shot Setup Reports: Script Editor

Report Generator  |  Recorder Menu  |  Config Vars  |  Report DTO Cache  |  Quartz  |  jFig  |  Test Report  |  Test XML  |  Test LPOM

Save  |  Copy  |  Delete  |  New  |  ?

Report: Target And Cryo

Parent: Target System
Report Name: Target And Cryo
HTML URL: TargetAndCryoHtml.action
On Menu: Yes

Version: 0
Report Alias: targetAndCryo
Excel URL: TargetAndCryoExcel.action

Content:

```html
#end ## GO THROUGH ALL LOCATIONS #if($locations) #set($cnt = 0) #foreach($loc in $locations.values()) #if($tsLocations[$loc]) #if($cnt > 0)

#end ### run ts report $report1(alias='targetSelectionLocation',title='Target Selection $loc',displayHeader=true,noHeader=true,customReportName='Target Selection $loc',loc=$loc,subReport=true,ajaxTemplate='progressbarAjaxTemplate') #set($cnt = $cnt +1) #end #if($cryoLocations[$loc]) #if($cnt > 0)

#end ### run cryo report $report1(alias='targetCryoLocation',title='Target Cryo $loc',displayHeader=true,noHeader=true,customReportName='Target Cryo $loc',loc=$loc,subReport=true,ajaxTemplate='progressbarAjaxTemplate') #set($cnt = $cnt+1) #end #end #end
```

Save  |  Preview Html  |  Preview Excel

2013-000000s2.ppt  
Name - ICALEPCS, 10/6/13