

adinstbpm - Task #17737

Milestone # 17719 (New): Booster operational test of one crate alongside existing BPM system

bpmd error facility

09/16/2017 10:43 AM - John Diamond

Status:	Work in progress	Start date:	09/16/2017
Priority:	Normal	Due date:	
Assignee:	John Diamond	% Done:	100%
Category:		Estimated time:	24.00 hours
Target version:		Spent time:	6.00 hours
Description			
Design and implement a smart error handling facility for bpmd.			

History

#1 - 09/16/2017 10:43 AM - John Diamond

- Tracker changed from Milestone to Task

#2 - 09/19/2017 06:36 PM - John Diamond

One idea is a family of C++ exceptions for specific BPM error conditions and a function for converting from a BPM exception (NO MEASUREMENT) to an ACNET error (BPM NO DATA).

#3 - 09/20/2017 03:21 PM - John Diamond

- Status changed from Assigned to Work in progress

- % Done changed from 0 to 50

Implemented an exception class called BPMError that ADInstBPM can use for throwing exceptions that identify a subsystem and error code much like ACNET/MOOC.

Defined several global constant BPMErrors in defs.H. Went through code line-by-line and replaced runtime_error throw's with the appropriate BPMError instance.

Modified ACNET classes to catch certain BPMErrors and re-throw the appropriate AcnetError that the front end framework uses to signal error conditions to ACNET. Also added several global constants that define various BPM error codes from the rbpm project that are already understood by ACNET.

We'll see if this system works well and will update this ticket with ideas for improvement.

#4 - 10/20/2017 01:56 PM - John Diamond

Seems to be adequate for now, bumping % done to 100% but keeping the status at "work in progress".

#5 - 10/20/2017 01:56 PM - John Diamond

- % Done changed from 50 to 100