



RECOVERY ACTIONS FOR ALARM AND WARN EVENTS

With EventSpring versions 2.4.05 and prior, only one recovery action (RA) could be defined, and it would be executed for both warnings and alarms. The RA needed logic to determine what to do based on the event state (if ALARM then do X else if WARN then do Y).

This limitation can often be worked around by using the `arsAction` rule, which can differentiate between alarms and warnings (for example, set `arsActionALARM=6` and `arsAction=4` so only ALARM conditions will trigger the RA.)

However, current versions of EventSpring (V2.4.05-02 and later) support specification of RAs based on alert status (for example, instead of just `arsCommand`, these variables have been added: `arsCommandALARM` and `arsCommandWARNING`.)

HOW EVENTSPRING PASSES DATA TO THE RA

Unlike a standard PATROL recovery action, which is executed under the parameter for which it was called, an EventSpring RA is executed under the EventSpring application class. This means that a typical PSL call like `get("name")` will behave differently than you might expect. Fortunately, EventSpring compensates for this by passing some added PSL variables behind-the-scenes. EventSpring will insert, at the beginning of the PSL RA code, the following variable assignments:

```
__appl_class__=<name of application class>;  
__instance__=<name of application instance>;  
__param__=<name of parameter>;  
__param_status__=<parameter status>;  
__param_value__=<parameter value>;  
__data__=<any data passed to AS_EVS(>;  
__nfile__=<name of output file used to store RA output if desired>;
```

These variables can be referenced from the RA script like any other PSL variable.

Has this article been helpful? Please let us know how we can improve the newsletter in the future by sending suggestions, comments or questions to articles@advantisms.com.com

For more information on the services offered by Advantis Management Solutions, Inc., please feel free to contact us by e-mail at info@advantisms.com.com or by phone at 617-233-4986.