SYSCOM Error Handling Framework

Introduction

The purpose of an advanced error handling mechanism, in the context of a workflow solution, is to monitor certain critical elements of the workflow system (and support systems used in activity implementations and process starters) and take the appropriate corrective actions (notification of support staff for example) when errors are identified.

The following types of applications should be part of an advanced Error Handling Framework:

- Scheduled, unattended workflow-driving applications:
  - Process starters
  - Maintenance utilities (garbage collection, worklist monitors)
  - Scheduled process terminators
- User applications:
  - GUI applications on FAT or THIN workflow clients
  - Invisible components used by GUI applications to perform workflow functions
- Automated workflow-driven applications
  - UPES-implemented workflow activities
  - PEA-implemented workflow activities

All automatic workflow activities should be tracked to ensure they are started, and completed within an acceptable amount of time. This tracking should be handled by a worklist-monitoring utility that would look for "first notifications" of automatic (unattended) activities. In order to track unattended activities, process modelers should follow the convention of specifying notifications for those unattended activities that do not complete within an appropriate timeframe.

The Error Handling Framework (EHF) provides a uniform approach to signaling and handling errors from all the application types listed above. Additionally, the Workitem Handling Facility can be combined with the EHF to track and report overdue workflow activities.

Functional overview

The following list summarizes the main features of the EHF:

- Java MQSeries based error-reporting client component. The client component ensures delivery of the (XML-based) error report. If errors occur during transfer of the error report, the report is saved on the client's file system and the transfer is invisibly retried until successful.
- Flexible formatting of the error report based on file templates. Error report formatting supports plain text and HTML error report templates
- Flexible sending capabilities. Final reporting of the error can be accomplished through an SMTP sender, MQSeries queues, or any other SYSCOM Message Handling Framework (MHF) sender node
- The SMTP sender node allows errors to be reported (in their formatted form) to various distribution lists, based on the application originating the error report
- Error report processing integrity: Errors occurring when the error message is processed/sent by the EHF are never lost (as long as the message serialization mechanism isn't compromised [e.g. file system full]) and can invisibly retried until the message is successfully reported to its final destination.
- EHF integration with the MHF as a logging service: This allows any MHF-based application component (e.g. UPES activity implementations) to report errors through the standard (non EHF-aware) MHF logging mechanism.
- EHF integration with the SYSCOM Workitem Handling Facility: This allows – for example – the Workitem Handling Facility to detect overdue workflow activities and report them to the EHF. This could in turn generate emails to report those overdue items.

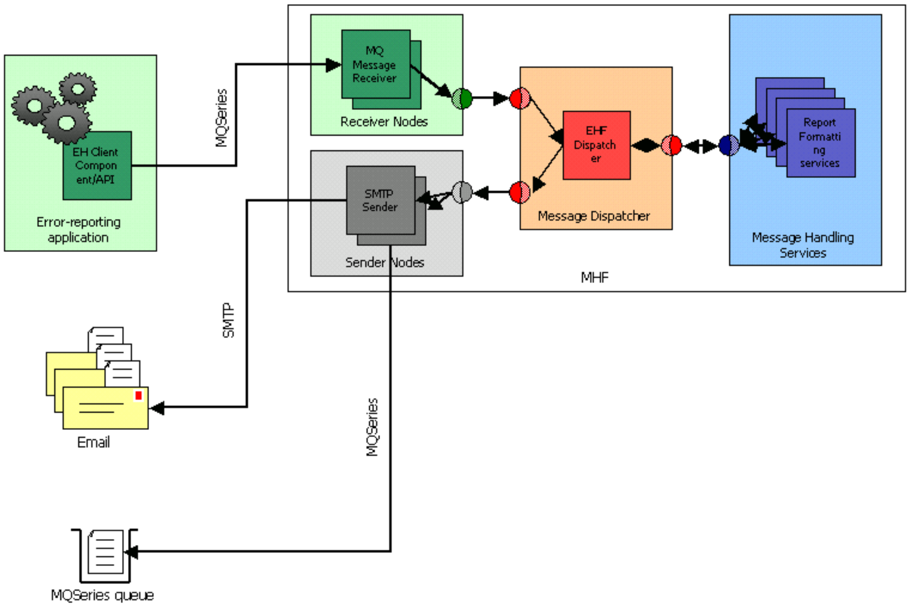
- EHF integration with the SYSCOM logging framework: This allows non-EHF aware applications that log through the SYSCOM logging framework to log errors that are eventually reported through the EHF.

High level framework design

The EHF is divided into 5 major functional areas:

1. The client component used to report the error to the central error handling mechanism
2. The receiver of the client error report
3. The dispatcher of the report to the proper report-formatting service
4. The report formatting service(s)
5. The sender(s) of the formatted report

The diagram below presents relationships between the 5 functional areas of the EHF as identified above.



In general, an application that detects an error and has access to the EHF client component can use the component to report the error condition. The following report fields can be set by the error-reporting application through the EHF client:

- Timestamp of the error
- ID of the reporting application
- Instance of the reporting application (this can be useful if several instances of the same application are running)
- Host information (e.g. IP address & hostname) of the host running the application
- Severity of the error
- Return code for the error (specified as a long)
- Textual description of the error

The report sent by the client is in XML format (that format is expected by the EHF in order to process the report correctly). Once the error is reported by the EHF client to the EHF, the Id of the reporting application is checked to see if a special reporting template should be used for the application. If not, a default template is used.

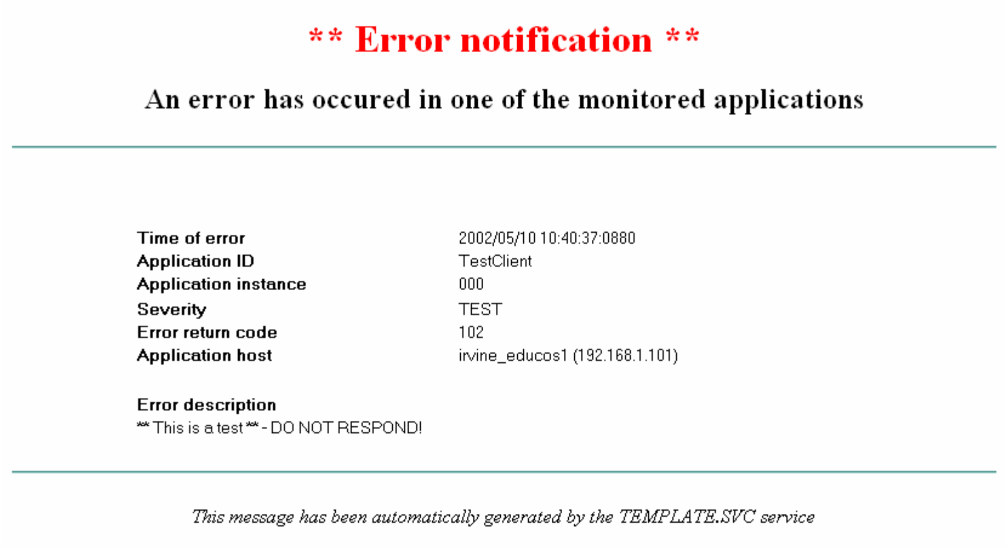
Reporting templates allow the XML-formatted report to be transformed into something else – usually (but not necessarily) a text or HTML-based version of the report.

The sample below gives an example of the original report and its transformed equivalent.

XML-formatted report – as sent by the EHF client:

```
<error_report>
  <timestamp>Fri May 10 2002 - 10:40:37:0880 AM</timestamp>
  <application>
    <id>TestClient</id>
    <ipaddr>irvine_educos1 (192.168.1.101)</ipaddr>
    <instance>000</instance>
  </application>
  <info>
    <rc>102</rc>
    <severity>TEST</severity>
    <errortext>** This is a test ** - DO NOT RESPOND!</errortext>
  </info>
</error_report>
```

Report transformed through the report-formatting service:



** Error notification **

An error has occured in one of the monitored applications

Time of error	2002/05/10 10:40:37:0880
Application ID	TestClient
Application instance	000
Severity	TEST
Error return code	102
Application host	irvine_educos1 (192.168.1.101)

Error description
** This is a test ** - DO NOT RESPOND!

This message has been automatically generated by the TEMPLATE.SVC service

Once the formatted report is created, it is sent by the EHF sender. If the sender user is the SMTP EHF sender, then different distribution lists (recipients, cc, bcc) can be specified based on the originating application ID.

The EHF currently supports two sender types:

- The SMTP sender (specific to the EHF)
- The MQSeries sender (available from the SYSCOM MHF)

Other template report formatters and report senders can be created through a simple API, made available through the SYSCOM MHF.

For more information…

If you would like additional information about the ***SYSCOM Error Handling Framework***, or, if you would like to engage SYSCOM in your project, please contact us at Sales@SYSCOM.com or:

Vickie Wysokinski
VP of Sales and Marketing
410.539.3737 x 1300
VWysokinski@SYSCOM.com

Rick Marcuson
Director of Sales – West
913.897.3304
RMarcuson@SYSCOM.com

Brian McConnell
Director of Sales – East
410.539.3737 x 1310
BMcConnell@SYSCOM.com