**UICDS WebEOC Adapter Design Description**

**High Level Requirements**

Goal: Create an adapter framework that can support multiple adapter components to different WebEOC boards. The framework shall support:

* two-way synchronization between boards
* one-way posting of data from a WebEOC server to a UICDS core
* one-way posting of data from a UICDS core to a WebEOC server
* configuration of adapters in a startup configuration file
* configuration of WebEOC server parameters in a startup configuration file
* configuration of UICDS core parameters in a startup configuration file

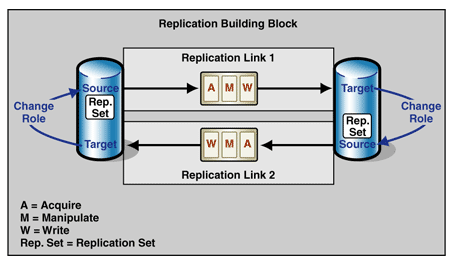
This document will describe two adapter components, the UICDS Incident Board and UICDS Position Log adapter components. The UICDS Incident Board is a two-way component and its description will illustrate the main parts of the overall WebEOC adapter design. The UICDS Position Log board is a one-way adapter and its description will focus on that adapters functionality.

**UICDS Incidents Board Adapter Component**

Goal: Create a UICDS/WebEOC adaptor for the WebEOC UICDS Incident board that provides two-way synchronization.

**Capability Description**

The WebEOC UICDS Incident Board adapter is a component that can be configured to run with the existing WebEOC adapter as specified in the Set Up section. The WebEOC adapter architecture basically follows the Master-Master Replication pattern documented at <http://msdn.microsoft.com/en-us/library/ff649910.aspx> without the conflict detection and resolution. The following diagram taken from the Microsoft site illustrates the software architecture.



In this instantiation of the pattern one source is the UICDS core and the other source is the WebEOC board. Both sources are polled for changes (Acquire), mapped into the appropriate fields for the target (Manipulate) and then posted to the target (Write). The following class diagram shows the classes involved in the WebEOC adapter for the UICDS Incident board.



In the class diagram the two replication links are implemented in the IncidentToUICDSIncidents and UICDSIncidentsBoardToIncident classes. The architecture is designed as an event driven system such that the main program in the WebEOCAdapter class is a loop that invokes a polling method on the WebEOCEventProducer and then the UicdsCore. These polling methods acquire data from their sources and produce events for the changed data to the appropriate listeners. The particular listeners then handle the events and perform the manipulate and write actions from the Master-Master pattern.

The IncidentToUICDSIncidents class implements a WorkProductListener and uses the UICDSIncidentsBoard class to represent the WebEOC board. The IncidentToUICDSIncidents processes each UICDS notification by creating or updating the corresponding WebEOC board entry using the WebEOCWebServiceClient as shown in the following sequence diagram.



The data is mapped from the UICDS Incident work product to the UICDS Incident Board fields as shown in the following table.

|  |  |
| --- | --- |
| UICDS Incident Work Product Element | WebEOC UICDS Position Log Board Field |
| ActivityName[0] | label |
| ActivityDescription[0] | description |
| ActivityIdentification[0].IdentificationID | uicdsid |
| ActivityDate[0] | event­\_datetime |
| IncidentLocation[0].LocationAddress[0] | address |
| If IncidentLocation[0].LocationArea.AreaCircularRegion. AreaPolygonGeographicCoordinate exists then use first point else use IncidentLocation[0].LocationArea.AreaCircularRegion. CircularRegionCenterCoordinate | latitude and longitude |
| ActivityCategory[0] | event\_type |
| IncidentJurisdictionalOrganization. OrganizationName | jurisdiction |
| IncidentJurisdictionalOrganization. rganizationPrincipalOfficial.PersonName. PersonFullName | contact\_name |

The UICDSIncidentsBoard class is used as a representation of the actual WebEOC board and as a cache of the entries that are currently on the board. The UICDSIncidentsBoardToIncident class maintains the cache by matching dataid fields from the WebEOC entries and setting the UICDS Incident ID field if it creates the UICDS incident. As the IncidentToUICDSIncidents class adds new entries to the board it sets the Incident ID of the cache entry. These entries are then used by the two classes to find the matching entry for any updates and the modified data/time for the WebEOC entry is used to determine which entries from the board have been updated.

The Incident ID value (from the Work Product Properties) is used to search the cached values for a match when processing a notification. If a match is not found the WebEOC server is queried directly using a filtered data request using the Incident ID value to match the uicdsid field in the UICDS Incidents Board.

When the adapter creates an entry on the UICDS Incidents board it adds an IncidentEvent to the UICDS Incident work product and updates the work product. The following is an example of an IncidentEvent element created by the adapter.

<IncidentEvent>

<ActivityIdentification>

<IdentificationID>1739</IdentificationID>

<IdentificationCategoryDescriptionText>demo.esi911.com/eoc7/api.aspx#UICDS Incidents</IdentificationCategoryDescriptionText>

</ActivityIdentification>

<ActivityCategoryText>WEBEOC</ActivityCategoryText>

<ActivityReasonText>Received</ActivityReasonText>

</IncidentEvent>

The combination of ActivityCategoryText and ActivityReasonText indicates that this event captures the fact that WebEOC received this incident. The data in the ActivityIdentification element indicates that the WebEOC server at demo.esi911.com/eoc7/api.aspx received the incident and created an entry with dataid 1739 in the UICDS Incidents board.

Errors received when communicating with the WebEOC server are handled in the IncidentToUICDSIncidents class by saving the Work Product in a queue. On the next polling action that queue will be processed first before any new notifications. The UICDSIncidentsBoardToIncident handles errors from the WebEOC server by not updating the cached entry and therefore on the next polling action the incoming WebEOC entry will be processed again.

**Limitations**

The adapter relies on the fact that the Position of the WebEOC login information to determine who posted the last change to a WebEOC entry. If a user logs in with that Position and makes changes to any UICDS Incidents Board entries those change will not be propagated back to the UICDS core. This is to stop infinite loops of changes.

The adapter relies on the fact that the UICDS resource instance name is being used only by this adapter. To stop infinite loops of changes it will not propagate any Incident work product notifications that have the UpdatedBy field set to the adapters UICDS resource instance to the WebEOC server.

**Set Up**

The WebEOC adapter configuration requires the following data to be specified in the webeoc-context.xml file. This file defaults to the one that is compiled into the executable jar but will use one in the local directory where the adapter is started if it exists.

1. The URL of the core to use
2. A username and password that is valid on that core
3. The URL of the WebEOC API
4. The name of the WebEOC incident to use
5. The username, password, and position to use for WebEOC

If it is possible it is best to remove the WebEOC incident and then recreate it on WebEOC before starting the adapter the first time that is the only way clear out all the old WebEOC entries. Remember that this also clears out all the entries for any board that has data for that WebEOC incident.

In general, a particular adapter component will require a Spring bean for the Java class that represents the WebEOC board and one for each replication link. The UICDS Incidents Board then requires three Spring beans, one for the UICDSIncidentsBoard, one for the IncidentToUICDSIncidents replication link, and one for the UICDSIncidentsBoardToIncident replication link.

The UICDSIncidentsBoard is configured in the webeoc-context.xml file using this bean:

<bean id="uicdsIncidentsBoard" class="com.saic.uicds.clients.em.webeocAdapter.UICDSIncidentsBoard">

<property name="incidentName" value="uicds test"/>

</bean>

The IncidentToUICDSIncidents WorkProductListener is configured in the webeoc-context.xml file using this bean:

<bean id=*"incidentToUicdsIncidents"* class=*"com.saic.uicds.clients.em.webeocAdapter.IncidentToUICDSIncidents"* init-method=*"initialize"*>

<property name=*"uicdsCore"* ref=*"uicdsCore"* />

<property name=*"webEocClient"* ref=*"webEOCWebServiceClient"* />

<property name=*"uicdsIncidentsBoard"* ref=*"uicdsIncidentsBoard"*/>

</bean>

The UICDSIncidentsBoardToIncident Board Listener is configured in the webeoc-context.xmjl file using this bean:

<bean id="uicdsIncidentsBoardToIncident" class="com.saic.uicds.clients.em.webeocAdapter.UICDSIncidentsBoardToIncident" init-method="initialize">

<property name="uicdsCore" ref="uicdsCore" />

<property name="id" value="IncidentsEvent1" />

<property name="webEOCEventProducer" ref="webEOCEventProducer" />

<property name="uicdsIncidentsBoard" ref="uicdsIncidentsBoard"/>

</bean>

**Operation**

The WebEOC adapter is a plain Java application packaged as an executable jar file. It can be executed on any command line on a machine where Java has been installed like:

> java –jar WebEOCAdapter.jar

**UICDS Position Log Adapter**

The IncidentToUICDSPositionLog class also pulls all the data from the UICDS Position Log board from WebEOC to maintain a cache of the current data. This cache is used during processing of work product updates to match current UICDS incidents with entries that already exist on the UICDS Position Log board. This data is pulled first before starting to handle notifications from the core during startup so that matching can take place if the adapter is restarting.

|  |  |
| --- | --- |
| UICDS Incident Work Product Element | WebEOC UICDS Position Log Board Field |
| ActivityName[0] | Name\_ |
| ActivityDescription[0] | Description |
| ActivityIdentification[0].IdentificationID | globalid |
| ActivityDate[0] | date\_time |
| IncidentLocation[0].LocationAddress[0] | Event\_Location |
| Last latitude and longitude from IncidentLocation[0].LocationArea | Appended to Event\_Location as (lat, long) |
| ActivityCategory[0] | Event\_Type |

The adapter attempts to match a UICDS Incident work product to a WebEOC Position Log entry in two ways. It first tries to find a match in the data included in the UICDS Incident work product contained in a IncidentEvent element and second it tries the cache of WebEOC Position Log entries cached from WebEOC. The first way of matching is the most reliable but cannot be used when processing a close and archive request because the full work product is no longer available.

When the adapter creates an entry on the UICDS Position Log board it adds an IncidentEvent to the UICDS Incident work prouduct and updates the work product. The following is an example of an IncidentEvent element created by the adapter.

<IncidentEvent>

<ActivityIdentification>

<IdentificationID>1739</IdentificationID>

<IdentificationCategoryDescriptionText>demo.esi911.com/eoc7/api.aspx#UICDS Position Log</IdentificationCategoryDescriptionText>

</ActivityIdentification>

<ActivityCategoryText>WEBEOC</ActivityCategoryText>

<ActivityReasonText>Received</ActivityReasonText>

</IncidentEvent>

The combination of ActivityCategoryText and ActivityReasonText indicates that this event captures the fact that WebEOC received this incident. The data in the ActivityIdentification element indicates that the WebEOC server at demo.esi911.com/eoc7/api.aspx received the incident and created an entry with dataid 1739 in the UICDS Position Log board.

The Event Identifier value (which is the Incident ActivityName) is used from the Digest contained in the notification to search the cached values for a match. Since UICDS allows multiple incidents to have the same ActivityName the code also matches against the description field.

**Limitations**

There is no way to mark a UICDS Position Board entry as closed. Currently the text “(CLOSED)” is prepended to the Description field in the WebEOC entry.

The IncidentToUICDSPositionLog class only pushes changes to the UICDS incident work product to the UICDS Position Log WebEOC board. Any changes to the entries on the board from a user logged into WebEOC will not get reflected back to the UICDS Incident work product and maybe overwritten if the associated work product changes on the UICDS core.

If the WebEOC user modifies the Name\_ or Description field on one of the records no more updates happen to the UICDS Incident work product until it is closed and archived then the adapter will not be able to correctly match the UICDS incident to the entry and it will not get tagged in the description as closed. This is because at the time of close the full Incident work product is no longer available so the UICDS Incident work product to WebEOC board entry matching has to take place through the cached data which will not produce a match.

The globalid field was used to hold the UICDS incident identifier but the purpose of that field when at use at MEMA is not clear. Therefore, the code was not written to rely on the UICDS incident identifier being available directly in the WebEOC record for matching purposes like it is for the UICDS Sig Event board.