

Connect5 Report Configuration

Project Implementation

Connect5 Design

Report Parameters

[Configuration 2](#_Toc356209054)

[XML Report Parameter Model 2](#_Toc356209055)

[Report Properties 2](#_Toc356209056)

[Web Reports 3](#_Toc356209057)

[HTML Report Screenshot 3](#_Toc356209058)

[HTML Report Search Screenshot 4](#_Toc356209059)

[Runtime Logging 5](#_Toc356209060)

[Database Logging Model 5](#_Toc356209061)

[XML Route Configuration 5](#_Toc356209062)

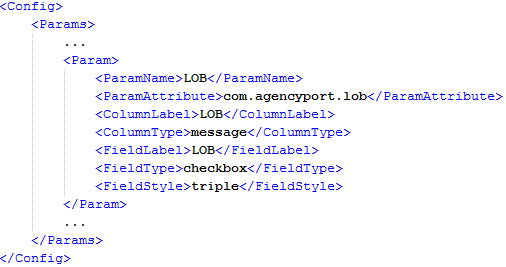
[Logging Properties 5](#_Toc356209063)

# Configuration

## XML Report Parameter Model

The configurations for the dynamic report parameters are stored in an XML file.

A partial sample can be seen below:

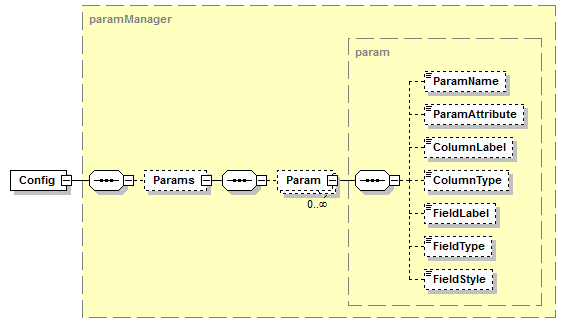


Search

Report

Logging

The actual schema looks like this (available in /web-inf/schema):



## Report Properties

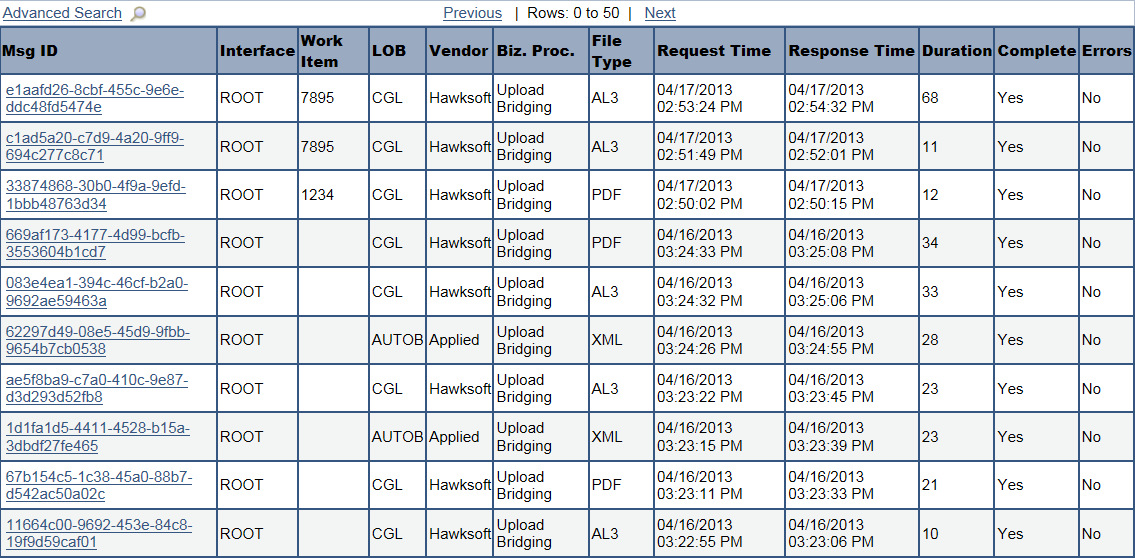
The following properties are used by the application:

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Default** |
| **report\_param.definition\_location** | Specifies location of Param XML configuration file. | ${context\_path}/WEB-INF/report/report\_param.xml |

# Web Reports

## HTML Report Screenshot

The message report can be seen below – the columns in red are dynamic.   
Only the parameters with a **Param.ColumnType** of “message” will be displayed, the **Param.ColumnLabel** is used as the header:

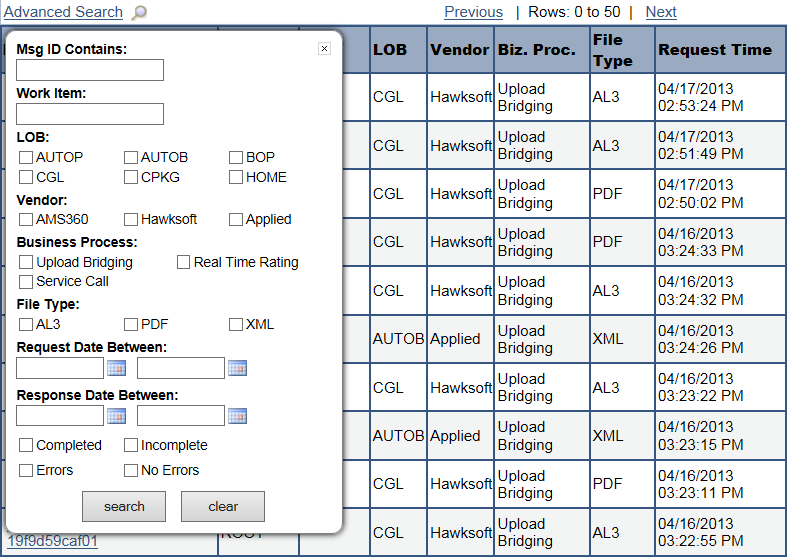


Report Parameters

## 

## HTML Report Search Screenshot

The message search can be seen below – the fields in red are dynamic.   
Only the parameters with a **Param.FieldType** of “text” or “checkbox” will be displayed, the **Param.FieldLabel** is used as the header and **Param.FieldStyle** is used to determine the number columns.   
The values are generated from distinct values in the database:

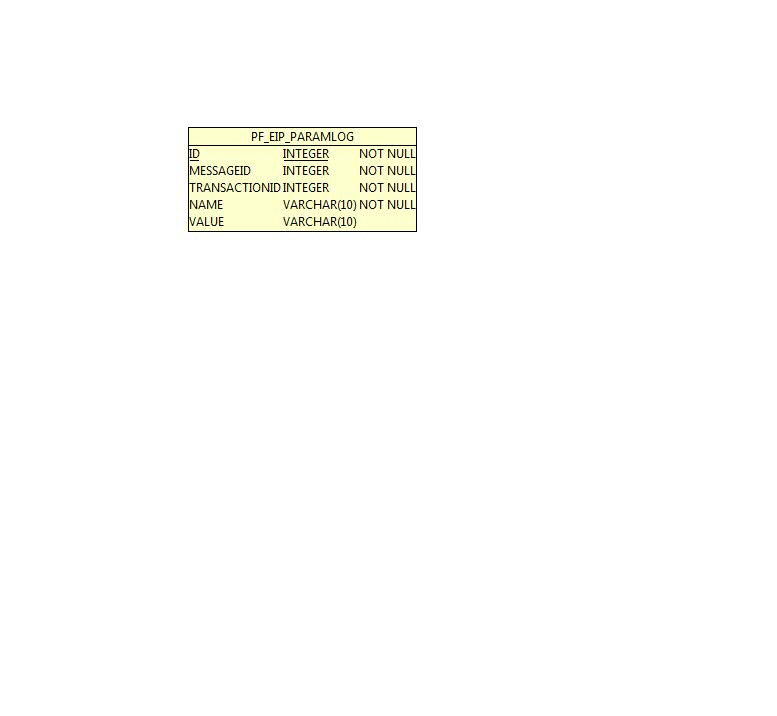


Report Parameters

# Runtime Logging

## Database Logging Model

The following table gets updated with the transaction data for each parameter. The ‘NAME’ column is populated from the **Param.ParamName** field and the ‘VALUE’ column is populated using the **Param.ParamAttribute** field to extract data from Transaction Attributes:



## XML Route Configuration

The attributes referenced above will need to be populated within your **route.xml** file – a sample is shown below with attributes in green:

<Processor

class="com.agencyport.substrate.module.APXPathAttributeProcessor"

name="Bridging XPath Attributes">

<ModuleConfig>

<ExecuteProcessor>true</ExecuteProcessor>

<AttributeScope>Transaction</AttributeScope>

<XPathExpressions>

[eip\_pair:**com.agencyport.producer**:eip\_name://Producer/ProducerInfo/ContractNumber:eip\_value]

[eip\_pair:**com.agencyport.insured**:eip\_name://InsuredOrPrincipal[InsuredOrPrincipalInfo/InsuredOrPrincipalRoleCd='Insured']/GeneralPartyInfo/NameInfo/CommlName/CommercialName:eip\_value]

[eip\_pair:**com.agencyport.lob**:eip\_name://CommlPolicy/LOBCd:eip\_value]

[eip\_pair:**com.agencyport.state**:eip\_name://CommlPolicy/ControllingStateProvCd:eip\_value]

[eip\_pair:**com.agencyport.premium**:eip\_name://CommlPolicy/CurrentTermAmt/Amt:eip\_value]

</XPathExpressions>

<Namespaces />

</ModuleConfig>

</Processor>

## Logging Properties

The following properties are used by the application:

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Default** |
| **com.pilotfish.eip.enableParamReporting** | Enables logging to the ParamLog database table. | true |