WebSphere (Ver 7)

Background

The WebSphere Server logs analysis App automatically Collect - Read - Parse - Analyzes - Reports all WebSphere machine generated log data of the server and presents a comprehensive set of graphs and reports to analyze machine generated data. Use a predefined set of dashboards and gadgets to visualize and address the system software, code written, and infrastructure during development, testing, and production. This WebSphere logs analysis App helps measure, troubleshoot, and optimize your servers integrity, stability and quality with visualization and investigation dashboards.

Steps

1. Add Log Data In XpoLog, When adding a log to XpoLog you can now set a Log Type (logtype). For WebSphere set the following logtypes for each log:
    a. **System out** - was,server,systemout
    b. **System err** - was,server,systemerr
    c. **Server start** - was,server,server-start
    d. **Server stop** - was,server,server-stop
    e. **Native out** - was,server,nativeout
    f. **Http error** - was,server,http-error
    g. **Http access** - was,server,access,w3c
2. In the WebSphere server configuration file, usually server.xml by default, located under the [SERVER_DIR]/config/.../[SERVER_NAME] directory. Search for the following parameters:
    a. **System out** - outputStreamRedirect
    b. **System err** - errorStreamRedirect
    c. **Server start** - outputStreamRedirect
    d. **Server stop** - outputStreamRedirect
    e. **Native out** - ioRedirect
    f. **Http error** - enableErrorLogging
    g. **Http access** - enableAccessLogging
3. Once the required information is set, on each log click next and edit the log pattern, this step is crucial to the accuracy and deployment of the Linux App. Use the following patterns for each of the logs:
    a. **System out - Basic Information -** [{date:Date,locale=en,MM/dd/yy HH:mm:ss:SSS z}] {text:Thread ID,charsLength=8;ftype=threadid;,} {text:Short Name,charsLength=13;ftype=shortname;,} {map:Event Type,ftype=severity;,F=FATAL;E=ERROR;W=WARNING;A=AUDIT;I=INFO;C=CONFIGURATION;D=DETAIL;O=SYSTEM OUTPUT;R=SYSTEM ERROR;Z=UNKNOWN}{block,start,emptiness=true} {text:Class,ftype=class;stopPattern=^com\\.ibm\\.[\\w\\.]+(\\s);,} {text:Method,ftype=method;,}{block,end,emptiness=true} {regexp:messagecode,refName=Message;ftype=messagecode,^\\s*([A-Z][A-Z][A-Z][A-Z]\\d\\d\\d\\d[EWI]):}{string:Message,ftype=message;,}
    b. **System out - Advanced Information -** [{date:Date,locale=en,MM/dd/yy HH:mm:ss:SSS z}] {text:Thread ID,charsLength=8;ftype=threadid;,} {map:Event Type,ftype=severity;,F=FATAL;E=ERROR;W=WARNING;A=AUDIT;I=INFO;C=CONFIGURATION;D=DETAIL;O=SYSTEM OUTPUT;R=SYSTEM ERROR;Z=UNKNOWN} UOW={text:UOW,ftype=uow;,} source={text:Source,ftype=source;,}{block,start,emptiness=true} class={text:Class,ftype=class;,} method={text:Method,ftype=method;,}{block,end,emptiness=true} org={text:Organization,ftype=organization;,} prod={text:Product,ftype=product;,} component={text:Component,ftype=component;,} thread=[{text:Thread Name,ftype=thread;,}]{regexp:messagecode,refName=Message;ftype=messagecode,^\\s*([A-Z][A-Z][A-Z][A-Z]\\d\\d\\d\\d[EWI]):}{string:Message,ftype=message;,}
    c. **System err -** [{date:Date,locale=en,MM/dd/yy HH:mm:ss:SSS z}] {text:Thread ID,charsLength=8;ftype=threadid;,} {text:Short Name,charsLength=13;ftype=shortname;,} {map:Event Type,ftype=severity;,F=FATAL;E=ERROR;W=WARNING;A=AUdit;I=INFO;C=CONFIGURATION;D=DETAIL;O=SYSTEM OUTPUT;R=SYSTEM ERROR;Z=UNKNOWN} {string:Message,ftype=message;,}
    d. **System start -** [{date:Date,locale=en,MM/dd/yy HH:mm:ss:SSS z}] {text:Thread ID,charsLength=8;ftype=threadid;,} {text:Short Name,charsLength=13;ftype=shortname;,} {map:Event Type,ftype=severity;,F=FATAL;E=ERROR;W=WARNING;A=AUDIT;I=INFO;C=CONFIGURATION;D=DETAIL;O=SYSTEM OUTPUT;R=SYSTEM ERROR;Z=UNKNOWN}{block,start,emptiness=true} {text:Class,ftype=class;stopPattern=^com\\.ibm\\.[\\w\\.]+(\\s);,} {text:Method,ftype=method;,}{block,end,emptiness=true} {regexp:messagecode,refName=Message;ftype=messagecode,^\\s*([A-Z][A-Z][A-Z][A-Z]\\d\\d\\d\\d[EWI]):}{string:Message,ftype=message;,}
    e. **System stop -** [{date:Date,locale=en,MM/dd/yy HH:mm:ss:SSS z}] {text:Thread ID,charsLength=8;ftype=threadid;,} {text:Short Name,charsLength=13;ftype=shortname;,} {map:Event Type,ftype=severity;,F=FATAL;E=ERROR;W=WARNING;A=AUDIT;I=INFO;C=CONFIGURATION;D=DETAIL;O=SYSTEM OUTPUT;R=SYSTEM ERROR;Z=UNKNOWN}{block,start,emptiness=true} {text:Class,ftype=class;stopPattern=^com\\.ibm\\.[\\w\\.]+(\\s);,} {text:Method,ftype=method;,}{block,end,emptiness=true}

{regexp:messagecode,refName=Message;ftype=messagecode,^\\s*([A-Z][A-Z][A-Z][A-Z]\\d\\d\\d\\d[EWI]):}{string:Message,ftype=message;,}

f. **Native out -** [{date:Date,locale=en,MM/dd/yy HH:mm:ss:SSS z}] {text:Thread ID,charsLength=8;ftype=threadid;,} {text:Short Name,charsLength=13;ftype=shortname;,} {map:Event Type,ftype=severity;,F=FATAL;E=ERROR;W=WARNING;A=AUDIT;I=INFO;C=CONFIGURATION;D=DETAIL;O=SYSTEM OUTPUT;R=SYSTEM ERROR;Z=UNKNOWN}{block,start,emptiness=true} {text:Class,ftype=class;stopPattern=^com\\.ibm\\.[\\w\\.]+(\\s);,} {text:Method,ftype=method;,}{block,end,emptiness=true} {regexp:messagecode,refName=Message;ftype=messagecode,^\\s*([A-Z][A-Z][A-Z][A-Z]\\d\\d\\d\\d[EWI]):}{string:Message,ftype=message;,}

g. **Http error -** [{date:Date,locale=en,EEE, dd MMM yyyy HH:mm:ss z}] [{priority:Severity,ftype=severity;,DEBUG;INFO;WARN;ERROR;CRITICAL}] [{geoip:Client IP,stopPattern=^[\d+:\.]+(:\d+/);ftype=remoteip;type=country:region:city}:{text:Remote Port,ftype=remoteport;,}/{text:Server Host,stopPattern=^[\d+:\.]+(:\d+\]);ftype=localip;,}:{text:Server Port,ftype=localport;,}] {string:Message,ftype=message;,}

h. **Http access - Basic Format -** {geoip:Client IP,ftype=remoteip;type=;,} {string:Remote Logical Username,ftype=remoteuser;,} {string:Remote User,ftype=remoteuser;,} [{date:Date,locale=en,dd/MMM/yyyy:HH:mm:ss z}] \"{choice:Method,ftype=reqmethod;,GET;POST} {string:URL,ftype=requrl;,}{block,start,emptiness=true}?{string:Query,ftype=querystring;,}{block,end,emptiness=true} {string:reqprotocol,ftype=reqprotocol;,}\" {number:Status,ftype=respstatus;,} {number:Bytes Sent,ftype=bytesent;,}{eoe}

i. **Http access - Combined Format -** {geoip:Client IP,ftype=remoteip;type=;,} {string:Remote Logical Username,ftype=remoteuser;,} {string:Remote User,ftype=remoteuser;,} [{date:Date,locale=en,dd/MMM/yyyy:HH:mm:ss z}] \"{choice:Method,ftype=reqmethod;,GET;POST} {string:URL,ftype=requrl;,}{block,start,emptiness=true}?{string:Query,ftype=querystring;,}{block,end,emptiness=true} {string:reqprotocol,ftype=reqprotocol;,}\" {number:Status,ftype=respstatus;,} {number:Bytes Sent,ftype=bytesent;,} \"{string:Referer,ftype=referer;,}\" \"{string:User Agent,ftype=useragent;,}\" \"{string:Cookie,ftype=cookie;,}\"{eoe}

System out Log Format Conversion Table

Format String	Description	XpoLog Pattern	XpoLog ftype
TimeStamp	The timestamp is formatted using the locale of the process where it is formatted. It includes a fully qualified date (for example YYMMDD), 24 hour time with millisecond precision and a time zone.	{date:Date,locale=en,MM/dd/yy HH:mm:ss:SSS z}	
ThreadId	An 8 character hexadecimal value generated from the hash code of the thread that issued the message.	{text:Thread ID,charsLength=8;ftype=threadid;,}	threadid
ShortName	The abbreviated name of the logging component that issued the message or trace event. This is typically the class name for WebSphere Application Server internal components, but can be some other identifier for user applications.	{text:Short Name,charsLength=13;ftype=shortname;,}	shortname
LongName	The full name of the logging component that issued the message or trace event. This is typically the fully qualified class name for WebSphere Application Server internal components, but can be some other identifier for user applications.	{text:Source,ftype=source;,}	source

EventType	A one character field that indicates the type of the message or trace event. Message types are in upper case. Possible values include: F A Fatal message. E An Error message. W A Warning message. A An Audit message. I An Informational message. C An Configuration message. D A Detail message. O A message that was written directly to System.out by the user application or internal components. R A message that was written directly to System.err by the user application or internal components. Z A placeholder to indicate the type was not recognized.	{map:Event Type,ftype=severity;,F=FATAL;E=ERROR;W=WARNING; A=AUdit;I=INFO;C=CONFIGURATION; D=DETAIL;O=SYSTEM OUTPUT;R=SYSTEM ERROR;Z=UNKNOWN}	severity
ClassName	The class that issued the message or trace event.	{text:Class,ftype=class;,}	class
MethodName	The method that issued the message or trace event.	{text:Method,ftype=method;,}	method
Organization	The organization that owns the application that issued the message or trace event.	{text:Organization,ftype=organization;,}	organization
Product	The product that issued the message or trace event.	{text:Product,ftype=product;,}	product
Component	The component within the product that issued the message or trace event.	{text:Component,ftype=component;,}	component