

# BlueStar Integration Server

build: 2018

# Outline

- What is BlueStar?
- Features
- Look Inside
- System Requirements

# What Is BlueStar?

- BlueStar Integration Server (BlueStar) is a platform designed for application integration and api gateway

# Features

- BlueStar Communication Service
  - UNC
  - FTP
  - Web Service (WS-Security) \*
  - Web API
  - WCF
  - SNA (LU0/APPC/DQ...)
  - TCPIP

*or other channel such as MSMQ, IBM MQ, TIBCO ESB... etc.*

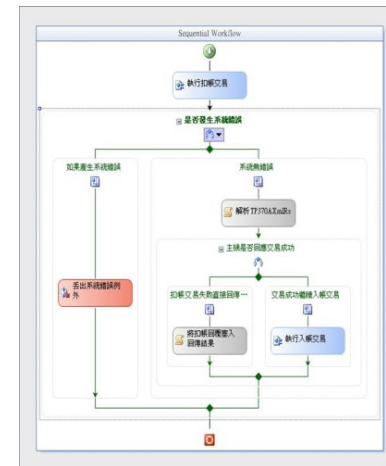
\* With built-in feature for transport authentication, message security and authorization

# Features

- BlueStar Transformation Service
  - High performance parser and schema cache mechanism
  - Legacy data type (ex. COBOL) conversion and validation
  - encode/decode with **surrogate** (4 bytes) support
  - transformation among JSON, XML and RAW Data (with various Encoding, such as:
    - EBCDIC
    - Unisys
    - ASCII, big5
    - Unicode)

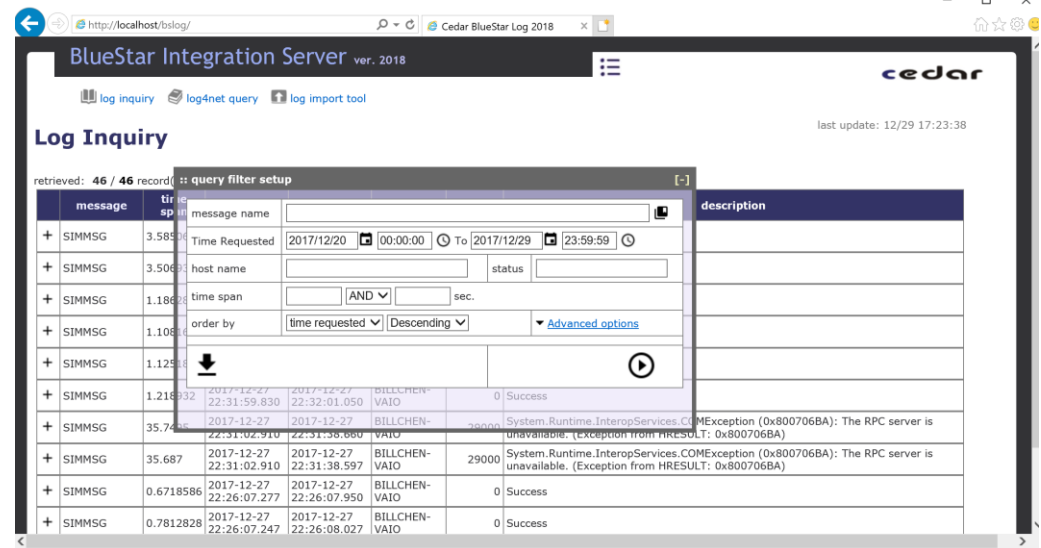
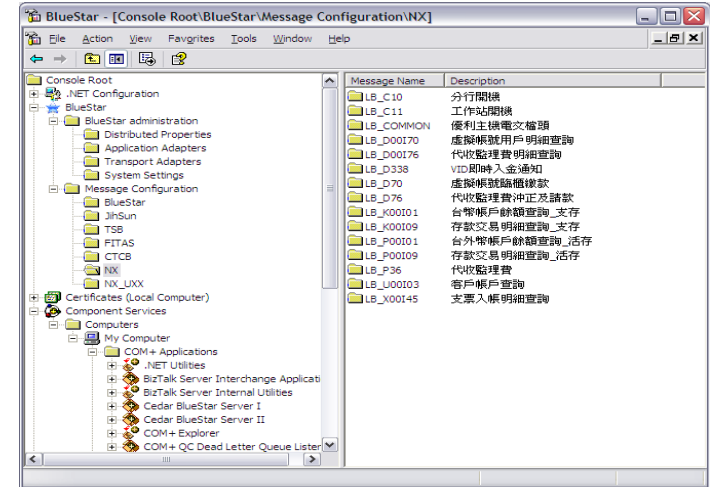
# Features

- BlueStar Workflow Service
  - Built-in workflow process engine for business transaction automation
  - Can be integrated with workflow solutions like BlueStar Safari, BPEL, Tibco, BEA, BizTalk, Workflow Foundation, ... etc.



# Features

- Microsoft Management Console
- Web-based Management And Monitoring Tool (Portal)



# Features

- Supports Visual Studio Development
  - Adapter/Workflow
  - Webservice processor/authorization plugin
  - Parser add-on
  - Custom codec ...
- Leverage log4net Logging Mechanism



# Look Inside

## BlueStar Admin Console

Monitor/shutdown/recycle the BlueStar adapters or components

The screenshot displays the BlueStar Admin Console interface. On the left, a tree view shows the hierarchy of components under 'COM+ Applications'. The right pane shows a real-time monitor table with columns for 'Prog ID', 'Objects', 'Activated', 'Pooled', 'In Call', and 'Call Time (ms)'. The table lists various components and their current status.

Prog ID	Objects	Activated	Pooled	In Call	Call Time (ms)
BAADevNull, DevNull					
BSAddOn, Util. 1					
BSDBLogger, Logger1					
BTFMAILC, XMLMail. 1					
CharsetConverter, Convert. 1					
CharsetConverter, Unisys. 1					
DispatchMsg, Select					
EAICodec, TGCodec	0	0		0	0
EAIInfo, Codec. 1					
EAIInfo, TypeCodec. 1					
eAIParser, TGParser					
FXMLMessage, Signature					
FXMLMessage, Validate					
FXMLServer, ProcessMessage					
HostFTP, MyFTP					
HTTPAdaptor, HTTPSend					
MailAdaptor, POP3Receive					
MailAdaptor, SMTPSend					
MC_SimSvc, BlueStarML					
MsgHandler, MCMsgHandler	0	0		0	0
MSMQAdaptor, MSMQSend					
NameSpace, NameData					
NameSpace, Params					
NameSpace, Server					

Allow start/shutdown/recycle BlueStar process and enable/disable application/component through COM+ MMC

Real time monitor with counter display

# Look Inside

BlueStar Admin Console  
Easy configuration for host  
transaction definition

The screenshot displays the BlueStar Admin Console interface. The main window is titled "ZBLIBL11 Properties" and is divided into several sections:

- Message configuration:** Includes fields for Group Name (CTCB), Message Name (ZBLIBL11), Description (支存餘額查詢), Type (HEX-EBCDIC), and Header (FNS\_HEADER).
- Transport:** Features a radio button for "PROGID" (selected) with the value POOLAPPC.HOSTAPPC, and an unchecked "Auto Response" option.
- Request Message Definition:** A list of fields for the incoming message:
  - 1: Name=COM\_BAL\_ID:Len=11:Type=X:Del=40
  - 2: Name=COM\_ACT\_NO:Len=16:Type=X:Del=4
  - 3: Name=COM\_BEGIN\_DATE:Len=8:Type=X:Del=
  - 4: Name=COM\_END\_DATE:Len=8:Type=X:Del=
  - 5: Name=COM\_QUEUE\_TYPE:Len=1:Type=X:Del=
- Receive Message Definition:** A list of fields for the outgoing message:
  - 1: Name=S\_MSG\_CODE:Len=4:Type=X:Del=1
  - 2: Name=S\_MSG\_TXO:Len=16:Type=X:Del=1
  - 3: Name=<INDICATOR:Len=1:Type=X:Del=1:N
  - 4: Name=S\_ID:Len=11:Type=X:Del=1
  - 5: Name=</S\_NAME:Len=42:Type=X:Del=1:Rej
  - 6: Name=S\_TX\_NO:Len=16:Type=X:Del=1
  - 7: Name=S\_ACT\_N

At the bottom of the window are "OK", "Cancel", and "Apply" buttons.

Message transformation  
editor

Message repository

Folder Name	Description
LB_D338	VID即時入金通知
LB_D70	虛擬帳號臨櫃繳款
LB_D76	代收監理費沖正及請款
LB_K00101	台幣帳戶餘額查詢_支存
LB_K00109	存款交易明細查詢_支存
LB_P00101	台幣帳戶餘額查詢_活存
LB_P00109	存款交易明細查詢_活存
LB_P36	代收監理費
LB_U00103	客戶帳戶查詢
LB_U00145	支票入帳明細查詢

# Look Inside

New surrogate codec

```
0" encoding="UTF-16"?>  
>004E</UNICODE>  
<AsciiEx>A447</AsciiEx>  
<EBCDIC937>1234</EBCDIC937>  
</Entry>  
- <Entry>  
  <!-- D85C DD44 -->  
  <!-- memory stream: 5c d8 44 dd -->  
  <UNICODE>夔</UNICODE>  
  <!-- 台 -->  
  <AsciiEx>A578</AsciiEx>  
  <EBCDIC937>6789</EBCDIC937>  
</Entry>  
- <Entry>  
  <!-- 鷓 D85C DD45-->  
  <UNICODE>5CD845DD</UNICODE>  
  <AsciiEx>A440</AsciiEx>  
  <!-- -0x4c4e is — in EBCDIC-->  
  <EBCDIC937>4c41</EBCDIC937>  
</Entry>  
</EUDM>
```

Unicode surrogate (4 bytes) support.

# Look Inside

Tool for AJAX Call with JSON Data Format

To convert JSON format to XML format message

To convert XML format message to JSON format

Textbox to type in your request message in JSON format

Textbox to type in your request message in XML format

Response message in JSON format

The screenshot displays the BlueStar AJAX Call Tool interface. At the top, there are tabs for 'demo', 'web service', and 'ajax call', with the 'cedar' logo on the right. The main area is divided into several sections:

- Request Message (JSON):** A large text area containing a JSON object: 

```
{ "SimpleTest": { "name": "John Doe", "message": "Hello, there!" } }
```

 Below this is a 'Submit' button and an 'OK' button.
- Request Message (XML):** A text area showing the XML equivalent of the JSON above: 

```
<BlueStar MsgName="SimpleTest"><name>John Doe</name><message>msgHello, there!</message></BlueStar>
```
- Response JSON:** A section titled 'response JSON:' containing a JSON object: 

```
{ "BlueStar": { "@xmlns": "http://www.cedar.com.tw/bluestar/", "@Status": "0", "Code": "OK", "output": "This is the output of AutoResponse", "timestamp": "2017-09-20 13:46:09.018", "MsgName": "SimpleTest", "RqUid": "fae06d0d-1060-4a50-8c59-4fad8cc4ce1e" } }
```

Annotations with red boxes and arrows point to the JSON request, XML request, and JSON response areas.

# System Requirements

- Windows Server 2016 and up
- .NET Framework 4.5.1