

# *Kako integrirati BI Publisher u Oracle Formse preko WEB Service-a?*

Mario Fistanić

Sektor za informatiku i telekomunikacije



# Sažetak



- BI Publisher vs Reports
- Web Service
- Implementacija

# Bi Publisher vs Reports



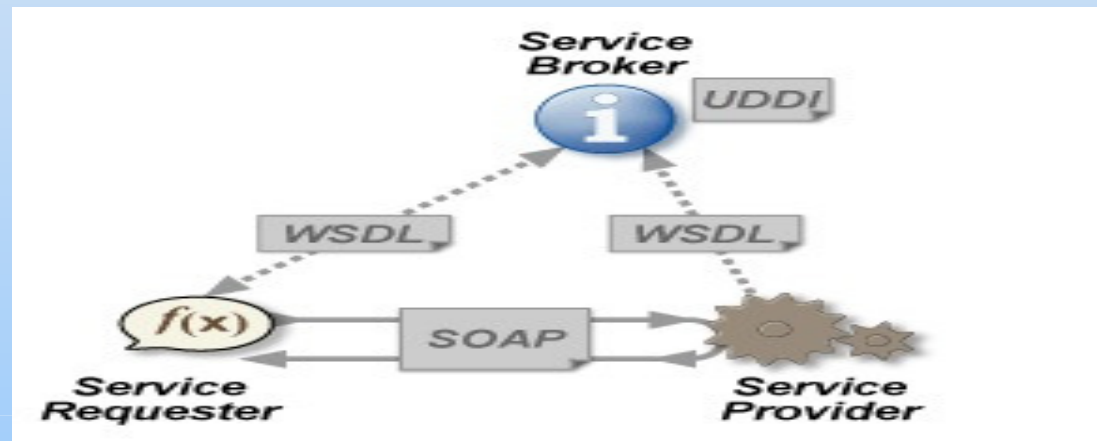
Zašto bi uopće koristili BI Publisher za izvještavanje a ne i dalje Reportse?

Je li BI Publisher bolji od Reportsa?

Koliko je teško integrirati ga u postojeća aplikativna rješenja?

U kojem smjeru ide Oracle?

# WEB Service



The [W3C](#) defines a "Web service" as:  
[...] a software system designed to support [interoperable](#) machine-to-machine interaction over a [network](#).

# Implementacija



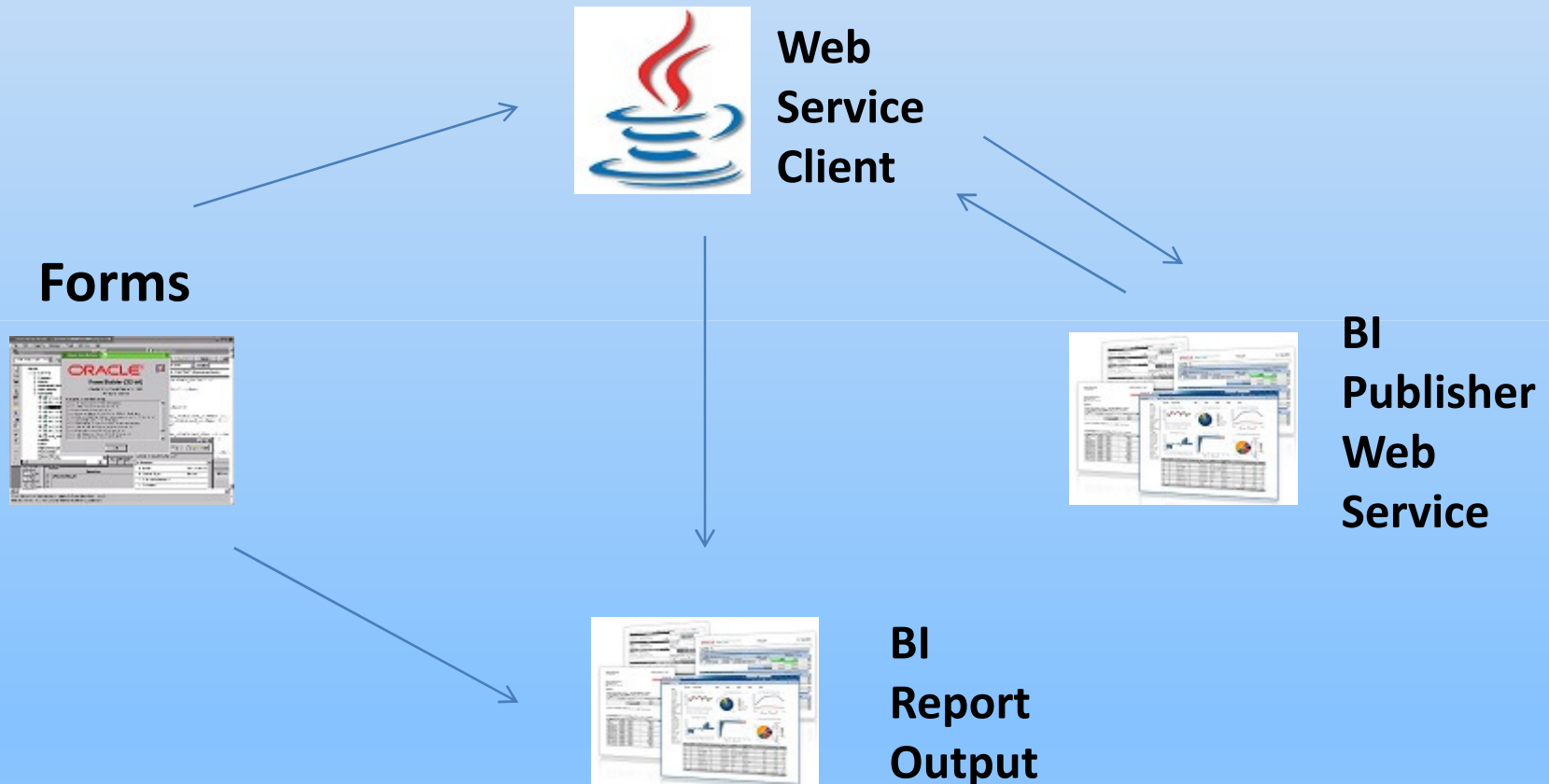
- Oracle Forms 10.1.2.3.0
- Oracle JDeveloper 11.1.2.3
- Oracle BI Publisher 11.1.1.6

- Glavna komponenta radi koje se može ostvariti komunikacija BI Publisher a i „ostatka svijeta ” (Formsi, ADF, APEX,...) je

## **BIP PUBLISHER PUBLIC WEB SERVICE API**

- Oracle BI Publisher 10.1.3.3.1 nadalje

# Integracija Oracle BI Publisher i Formsa



# Integracija Oracle BI Publisher a Formsa



1. kreirat Java Web Service Client komponentu
  2. kreirat PL/SQL interface za komunikaciju sa Java Web Service Clientom
  3. definirat kod za zvanje BI Reporta u formi
  4. konfigurirat formsweb.cfg i ENV file
- Pozvat BI Report iz Formsa 😊

# Kreiranje Java Web Service Client komponente



- U Oracle JDeveloper-u 11.1.2.3
  - Java Desktop Application ( Application Name – Project Name)
  - Business Tier - Web services – Web service and Proxy – JAX vs Style
  - WSDL URL (copy WSDL into project) :  
<http://machine:<port>/xmlpserver/services/v2/ReportService?wsdl>
  - Package name
  - Don't generate any asynchronous methods
  
- New General – Deployment Profile – JAR File – naziv (deployamo jar file na file sistem )



# PublicReportServiceClient



```
package bip_webservice.proxy;

import java.io.FileOutputStream;

import oracle.webservices.transport.ClientTransport;
import oracle.webservices.OracleStub;
import javax.xml.rpc.ServiceFactory;
import javax.xml.rpc.Stub;

public class PublicReportServiceClient {
    private bip_webservice.proxy.PublicReportService _port;
    private String username;
    private String password;
    private String reportPath;
    private String format;
    private String template;
    private String locale;
    private int sizeOfDataChunkDownload;
    private String paramName;
    private String paramValue;
    private String outFile;

    public PublicReportServiceClient() throws Exception {
        ServiceFactory factory = ServiceFactory.newInstance();
        _port = ((bip_webservice.proxy.PublicReportServiceService)factory.loadService(bip_webservice.proxy.PublicReportServiceService.class)).getPublicReportService();
        username=null;
        password=null;
        reportPath=null;
        format=null;
        template=null;
        locale=null;
        sizeOfDataChunkDownload=-1;
        paramName=null;
        paramValue=null;
        outFile=null;
    }
}
```

# Metoda callRunReport



```
public void callRunReport (String reportPath, String[] paramName, String[] paramValue, String
username, String password, String format, String template, String outFile)
{
    try {
        bip_webservice.proxy.PublicReportServiceClient myPort =
        new bip_webservice.proxy.PublicReportServiceClient();
        // Calling runReport
        ReportRequest repRequest = new ReportRequest();
        repRequest.setReportAbsolutePath(reportPath);
        repRequest.setAttributeTemplate(template);
        repRequest.setAttributeFormat(format);
        repRequest.setAttributeLocale("hr-HR");
        repRequest.setSizeOfDataChunkDownload(-1);
        if (paramName != null)
        {
            ParamNameValue[] paramNameValue = new ParamNameValue[paramName.length];
            String[] values = null;
            for (int i=0; i<paramName.length; i++)
            {
                paramNameValue[i] = new ParamNameValue();
                paramNameValue[i].setName(paramName[i]);
                values = new String[1];
                values[0] = paramValue[i];
                paramNameValue[i].setValues(values);
            }
            repRequest.setParameterNameValues(paramNameValue);
        }
        else
            repRequest.setParameterNameValues(null);
        ReportResponse repResponse = new ReportResponse();
        repResponse = myPort.runReport(repRequest,username,password);
        byte[] baReport = repResponse.getReportBytes();
        FileOutputStream fio = new FileOutputStream(outFile);
        fio.write(baReport);
        fio.close();
    } catch (Exception ex) {
        ex.printStackTrace();
    }
}
```

# Kreiranje PL/SQL interfacea za komunikaciju sa Java Web Service Clientom



Preko Java Importera importamo sljedeće klase:

- *bip\_webservice.proxy.ReportServiceClient*
- *bip\_webservice.proxy.types.ArrayOfParamNameValue*
- *bip\_webservice.proxy.types.ArrayOfString*
- *bip\_webservice.proxy.types.ParamNameValue*
- *java.lang.String*
- *java.util.List*

# Bazne tablice



```
CREATE TABLE BIP_PARAMETRI
(
  BIP_REPORT_NAME      VARCHAR2 (60 BYTE)          NOT NULL,
  BIP_PARAM_NO         INTEGER                    NOT NULL,
  BIP_PARAM_NAME       VARCHAR2 (90 BYTE)         NOT NULL,
  BIP_PARAM_TYPE       VARCHAR2 (30 BYTE),
  BIP_PARAM_PROMPT     VARCHAR2 (100 BYTE),
  BIP_NULL_ALLOWED     VARCHAR2 (3 BYTE),
  BIP_PARAM_DEFAULT    VARCHAR2 (90 BYTE),
  BIP_PARAM_HELP_MSG   VARCHAR2 (210 BYTE),
  BIP_PARAM_ERROR_MSG  VARCHAR2 (210 BYTE),
  BIP_LOV_FORM         VARCHAR2 (60 BYTE),
  BIP_GLOBAL_VAR       VARCHAR2 (210 BYTE)
)
```

```
CREATE TABLE BIP_REPORTI
(
  BIP_NAME             VARCHAR2 (60 BYTE)          NOT NULL,
  BIP_TITLE            VARCHAR2 (200 BYTE),
  BIP_PATH             VARCHAR2 (200 BYTE),
  BIP_FORMAT           VARCHAR2 (18 BYTE),
  BIP_TEMPLATE         VARCHAR2 (200 BYTE),
  BIP_OUT_PATH         VARCHAR2 (200 BYTE),
  BIP_WEBSHOW          VARCHAR2 (200 BYTE),
  BIP_SORT_NO         NUMBER
)
```

```
CREATE TABLE BIP_FORMATI
(
  FORMAT_NAME         VARCHAR2 (20 BYTE),
  FORMAT_BIP          VARCHAR2 (10 BYTE),
  FORMAT_EXT          VARCHAR2 (5 BYTE)
)
```

# Kod za poziv BI Reporta



## When-button-pressed trigger:

```
obj := PUBLICREPORTSERVICECLIENT.new();

un := 'Mario';
pw := 'Fistanic';

select dbms_random.string('L', 10) into random_alfa from dual;
select dbms_random.value(1,1000) into random_num from dual;

if :ext is null then
    select format_ext into :ext from bip_formati
    where format_name = :bip_format;
end if;

:out_file_name:= random_alfa||random_num||'.'||:ext;
out_file:= :bip_out_path||:out_file_name;

:out_file := out_file;

set_param_jarray (arr_name, arr_value);

n:= ora_java.get_array_length(arr_name);
n_1 := ora_java.get_array_length(arr_value);

For i in 1..n loop
    value_1 := ORA_JAVA.GET_STRING_ARRAY_ELEMENT(arr_name, i-1);
    value_2 := ORA_JAVA.GET_STRING_ARRAY_ELEMENT(arr_VALUE, i-1);

END LOOP;

publicreportserviceclient.callRunReport(obj,:bip_path,arr_name,arr_value,un,pw,nvl(:format_publisher,:bip_format),:bip_template,out_file);
web.show_document(:bip_webshow||:out_file_name);
```

# Konfiguracija formsweb.cfg i ENV file

- formsweb.cfg
  - C:%ORACLE\_HOME%\forms\server – definiramo ENV file za svaku aplikaciju posebno
- ENV file (aplikacije)
  - u njemu definiramo JAR fileove potrebne za rad aplikacije; među njima i naš kreirani JAR

BIP over URL???

# Q & A????

Interaktivnost  
reporta?

**HVALA NA PAŽNJI**